HEIDENHAIN

User's Manual ISO Programming

TNC 360

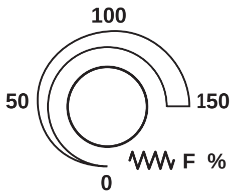
February 1994

Keys and Controls on the TNC 360

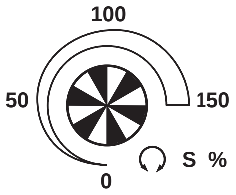
Controls on the Visual Display Unit

☼	Brightness
◑	Contrast

Override Knobs



Feed rate



Spindle speed

Machine Operating Modes

✋	MANUAL OPERATION
⎈	ELECTRONIC HANDWHEEL
⇥	POSITIONING WITH MANUAL DATA INPUT
⇥	PROGRAM RUN, SINGLE BLOCK
⇥	PROGRAM RUN, FULL SEQUENCE

Programming Modes

⇥	PROGRAMMING AND EDITING
⇥	TEST RUN

Program and File Management

PGM NR	Select programs and files
CL PGM	Delete programs and files
PGM CALL	Enter program call in a program
EXT	External data transfer
MOD	Supplementary modes

Moving the Cursor and Selecting Blocks, Cycles and Parameter Functions with GOTO

← ↑ → ↓	Move the cursor (highlight)
GOTO □	Go directly to blocks, cycles and parameter functions

Graphics

MOD	Graphic operating modes
BLK FORM	Define blank form, reset blank form
MAGN	Magnify detail
START	Start graphic simulation

Address Letters for ISO Programming

N	Block number
G	G function
F	Feed rate / Dwell time with G04 / Scaling factor
M	Miscellaneous function (M function)
S	Spindle speed in rpm
D	Parameter definition
H	Polar angle/Rotation angle in cycle G73
I J K	X, Y, Z coordinate of circle center/pole
L	Assign a label number with G98/ Jump to a label number/ Tool length with G99
R	Polar radius/ Rounding radius with G25, G26, G27 Chamfer with G24 Circle radius with G02, G03, G05 Tool radius with G99
T	Tool definition with G99/ Tool call
TOUCH PROBE	Set a datum with the 3D touch probe system

Entering Numbers and Coordinate Axes, Editing

X ... IV	Select or enter coordinate axes in a program
0 ... 9	Numbers
.	Decimal point
+/–	Algebraic sign
⌖	Actual position capture
NO ENT	Ignore dialog queries, delete words
ENT	Confirm entry and resume dialog
END □	Conclude block
CE	Clear numerical entry or TNC message
DEL □	Abort dialog; delete program sections

TNC Guideline:
From workpiece drawing to program-controlled machining

Step	Task	TNC operating mode	Refer to Section
	Preparation		
1	Select tools	——	——
2	Set workpiece datum for coordinate system	——	——
3	Determine spindle speeds and feed rates	——	11.4
4	Switch on machine	——	1.3
5	Traverse reference marks	✋ or ☸	1.3, 2.1
6	Clamp workpiece	——	——
7	Set the datum / Reset position display ...		
7a	... **with** the 3D touch probe	✋ or ☸	2.5
7b	... **without** the 3D touch probe	✋ or ☸	2.3
	Entering and testing part programs		
8	Enter part program or download over external data interface	◈ or EXT	5 to 8 or 10
9	Test part program for errors	⇥	3.1
10	Test run: Run program block by block without tool	⊡	3.2
11	If necessary: Optimize part program	◈	5 to 8
	Machining the workpiece		
12	Insert tool and run part program	⇥	3.2

Sequence of Program Steps

Milling an outside contour

Programming step	Key/Function	Refer to Section
1 Create or select program		4.4
Input: Program number	PGM NR	
Unit of measure for programming		
2 Define workpiece blank for graphic display	G30/G31	4.4
3 Define tool(s)	G99	4.2
Input: Tool number	T...	
Tool length	L...	
Tool radius	R...	
4 Call tool data	T...	4.2
Input: Tool number		
Spindle axis	G17	
Spindle speed	S...	
5 Tool change		
Input: Feed rate (rapid traverse)	G00	e.g. 5.4
Radius compensation	G40	
Coordinates of the tool change position	X... Y... Z...	
Miscellaneous function (tool change)	M06	
6 Move to starting position		5.2/5.4
Input: Feed rate (rapid traverse)	G00	
Coordinates of the starting position	X... Y...	
Radius compensation	G40	
Miscellaneous function (spindle on, clockwise)	M03	
7 Move tool to (first) working depth		5.4
Input: Feed rate (rapid traverse)	G00	
Coordinate of the (first) working depth	Z...	
8 Move to first contour point		5.2/5.4
Input: Linear interpolation	G01	
Radius compensation for machining	G41/G42	
Coordinates of the first contour point	X... Y...	
Machining feed rate	F...	
if desired, with smooth approach: program G26 after this block		
9 Machining to last contour point		5 to 8
Input: Enter all necessary values for each contour element		
if desired, with smooth departure: program G27 after the last radius-compensated block		
10 Move to end position		5.2/5.4
Input: Feed rate (rapid traverse)	G00	
Cancel radius compensation	G40	
Coordinates of the end position	X... Y...	
Miscellaneous function (spindle stop)	M05	
11 Retract tool in spindle axis		5.2/5.4
Input: Feed rate (rapid traverse)	G00	
Coordinate above the workpiece	Z...	
Miscellaneous function (end of program)	M02	
12 End of program		

How to use this manual

This manual describes functions and features available on the TNC 360 from NC software number 259 900 08.

This manual describes all available TNC functions. However, since the machine builder has modified (with machine parameters) the available range of TNC functions to interface the control to his specific machine, this manual may describe some functions which are not available on your TNC.

TNC functions which are not available on every machine are, for example:

- Probing functions for the 3D touch probe system
- Rigid tapping

If in doubt, please contact the machine tool builder.

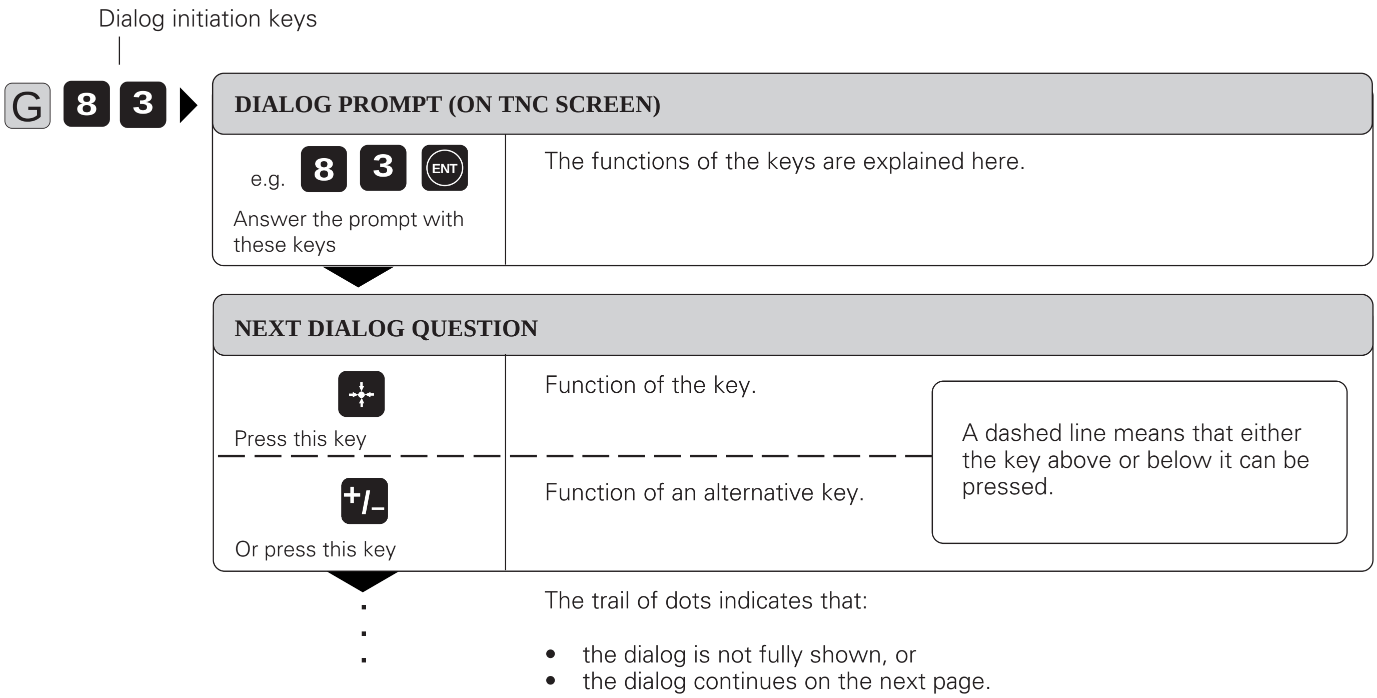
TNC programming courses are offered by many machine tool builders as well as by HEIDENHAIN. We recommend these courses as an effective way of improving your programming skill and sharing information and ideas with other TNC users.

The **TNC beginner** can use the manual as a workbook. The first part of the manual deals with the basics of NC technology and describes the TNC functions. It then introduces the techniques of conversational programming. Each new function is thoroughly described when it is first introduced, and the numerous examples can be tried out directly on the TNC. The TNC beginner should work through this manual from beginning to end to ensure that he is capable of fully exploiting the features of this powerful tool.

For the **TNC expert**, this manual serves as a comprehensive reference work. The table of contents and cross references enable him to quickly find the topics and information he needs. Easy-to-read dialog flowcharts show him how to enter the required data for each function.

The dialog flow charts consist of sequentially arranged instruction boxes. Each key is illustrated next to an explanation of its function to aid the beginner when he is performing the operation for the first time. The experienced user can use the key sequences illustrated in the left part of the flowchart as a quick overview. The TNC dialogs in the instruction boxes are always presented on a gray background.

Layout of the dialog flowcharts


Dialog initiation keys
G 8 3
DIALOG PROMPT (ON TNC SCREEN)
e.g. 8 3 ENT
Answer the prompt with these keys
The functions of the keys are explained here.
NEXT DIALOG QUESTION
Press this key
Function of the key.
+/–
Or press this key
Function of an alternative key.
A dashed line means that either the key above or below it can be pressed.
The trail of dots indicates that:
• the dialog is not fully shown, or
• the dialog continues on the next page.

Contents User's Manual TNC 360 ISO Programming

1 Introduction

2 Manual Operation and Setup

3 Test Run and Program Run

4 Programming

5 Programming Tool Movements

6 Subprograms and Program Section Repeats

7 Programming with Q Parameters

8 Cycles

9 External Data Transfer

10 MOD Functions

11 Tables, Overviews, Diagrams

1.1 The TNC 360

Control

The TNC 360 is a shop-floor programmable contouring control for milling machines, boring machines and machining centers with up to four axes. The spindle can be rotated to a given angular stop position (oriented spindle stop).

Visual display unit and operating panel

The monochrome screen clearly displays all information necessary for operating the TNC. In addition to the CRT monitor (BE 212), the TNC 360 can also be used with a flat luminescent screen (BF 110). The keys on the operating panel are grouped according to their functions. This simplifies programming and the application of the TNC functions.

Programming

The TNC 360 is programmed in ISO format. Programming with the easy to understand HEIDENHAIN plain language dialog format is also possible and is described in the TNC 360 User's Manual for HEIDENHAIN Conversational Programming.

Graphics

The graphic simulation enables you to test programs before actual machining. Various types of graphic representation can be selected.

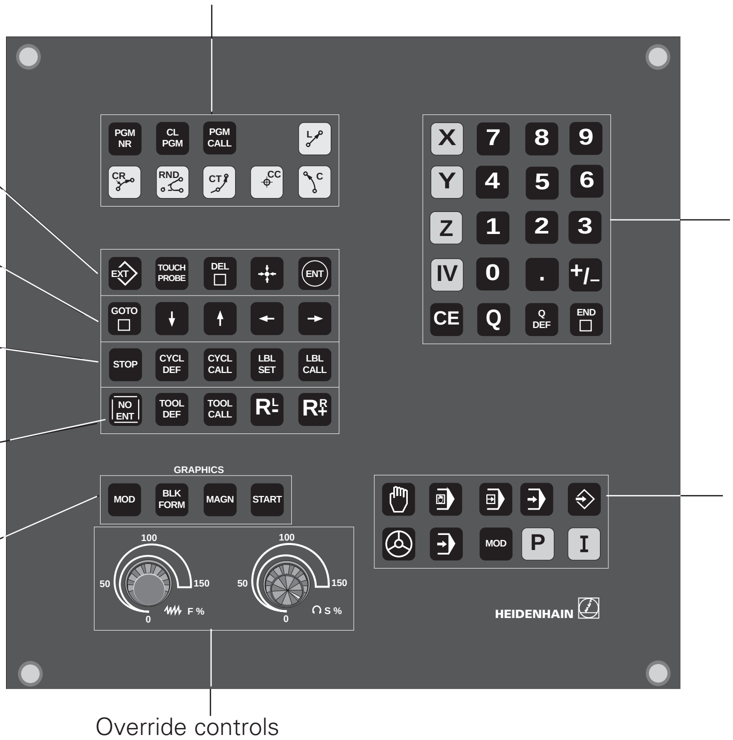
Compatibility

The TNC 360 can execute any part program that was programmed on a TNC 150B HEIDENHAIN control or any subsequent version.

The Operating Panel

The keys on the TNC operating panel are grouped according to their functions:

- Program selection
- Address letters



- External data transfer
- Probing functions
- Editing functions

- Jump instruction GOTO
- Arrow keys

- Address letters

- NO ENT key
- Tool-related address letters

Graphic operating modes

- Numerical entries
- Axis selection

Operating modes

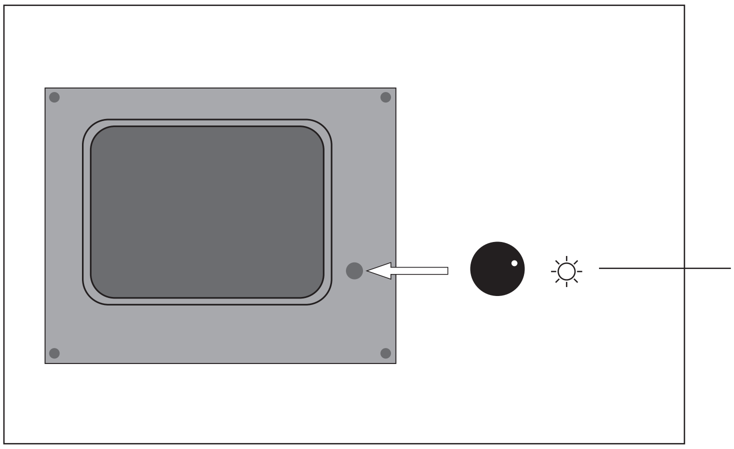
Override controls for spindle speed and feed rate

The functions of the individual keys are described on the inside front cover. An overview of the address letters used for ISO programming is provided in Chapter 11.

The machine operating buttons, such as I for NC start, are described in the manual for your machine tool.

In this manual they are shown in gray.

The Screen



Brightness control
(BE 212 only)

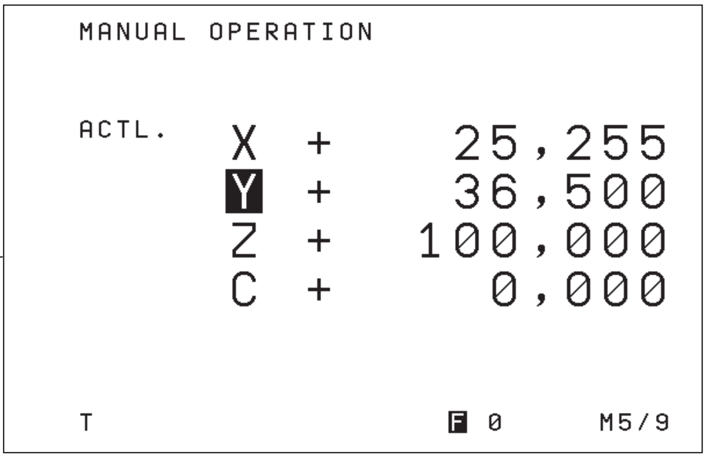
Header

The header of the screen shows the selected operating mode. Dialog questions and TNC messages also appear there.

Screen Layout

MANUAL and EL. HANDWHEEL operating modes:

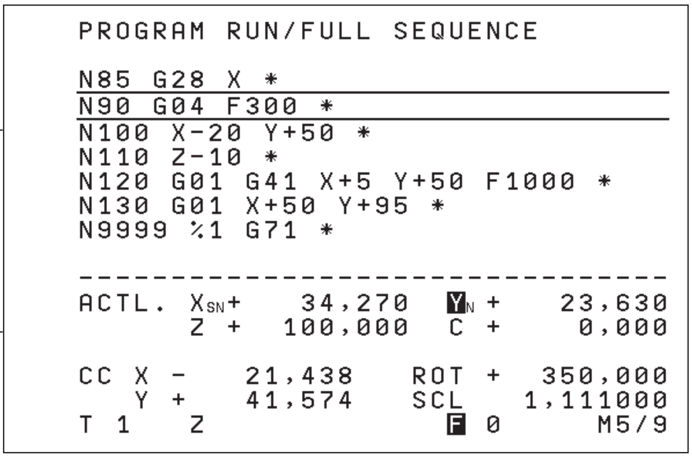
A machine operating mode has been selected



```
MANUAL OPERATION


ACTL.   X  +     25,255
        Y  +     36,500
        Z  +    100,000
        C  +      0,000



T                    F 0     M5/9
```

- Coordinates
- Selected axis
- ❊ means: control is in operation
- Status display, e.g. feed rate F, miscellaneous function M

A program run operating mode has been selected



```
PROGRAM RUN/FULL SEQUENCE

N85 G28 X *
N90 G04 F300 *
N100 X-20 Y+50 *
N110 Z-10 *
N120 G01 G41 X+5 Y+50 F1000 *
N130 G01 X+50 Y+95 *
N9999 %1 G71 *

--------------------------------
ACTL. XSN+   34,270  YN +   23,630
      Z +  100,000   C +    0,000

CC X -   21,438   ROT +  350,000
   Y +   41,574   SCL   1,111000
T 1  Z                F 0    M5/9
```

Section of selected program

Status display

The screen layout is the same in the operating modes PROGRAM RUN, PROGRAMMING AND EDITING and TEST RUN. The current block is shown between two horizontal lines.

TNC Accessories

3D Touch Probe Systems

The TNC features the following functions for the HEIDENHAIN 3D touch probe systems:

- Automatic workpiece alignment (compensation of workpiece misalignment)
- Datum setting
- Measurements of the workpiece can be performed during program run
- Digitizing 3D forms (optional, only available with HEIDENHAIN plain language dialog programming)

The TS 120 touch probe system is connected to the control via cable, while the TS 510 communicates by means of infrared light.

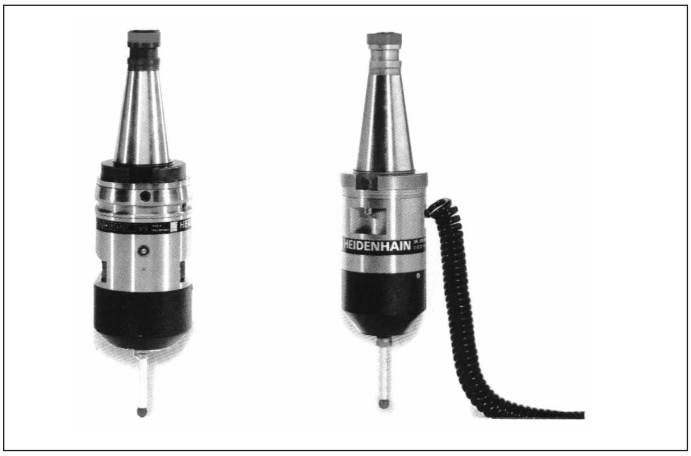


Fig. 1.5: HEIDENHAIN 3D Touch Probe Systems TS 120 and TS 511

Floppy Disk Unit

The HEIDENHAIN FE 401 floppy disk unit serves as an external memory for the TNC, allowing you to store your programs externally on diskette.

The FE 401 can also be used to transfer programs that were written on a PC into the TNC. Extremely long programs which exceed the TNC's memory capacity are “drip fed” block by block. The machine executes the transferred blocks and erases them immediately, freeing memory for further blocks from the FE.

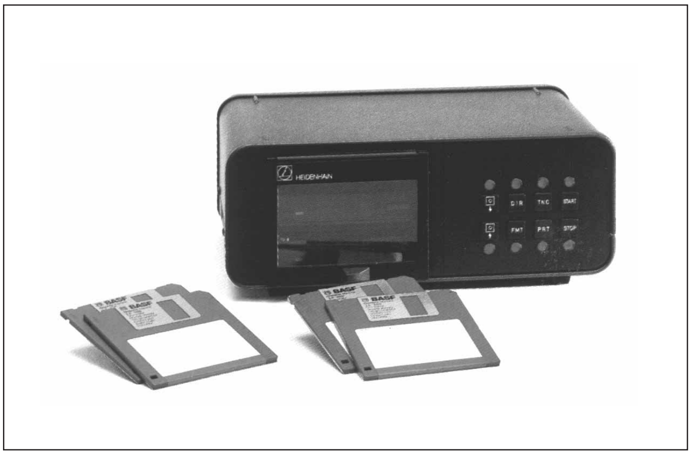


Fig. 1.6: HEIDENHAIN FE 401 Floppy Disk Unit

Electronic Handwheels

Electronic handwheels provide precise manual control of the axis slides. As on conventional machines, turning the handwheel moves the axis by a defined amount. The traverse distance per revolution of the handwheel can be adjusted over a wide range.

Portable handwheels, such as the HR 330, are connected to the TNC by cable. Built-in handwheels, such as the HR 130, are built into the machine operating panel.

An adapter allows up to three handwheels to be connected simultaneously. Your machine tool builder can tell you more about the handwheel configuration of your machine.

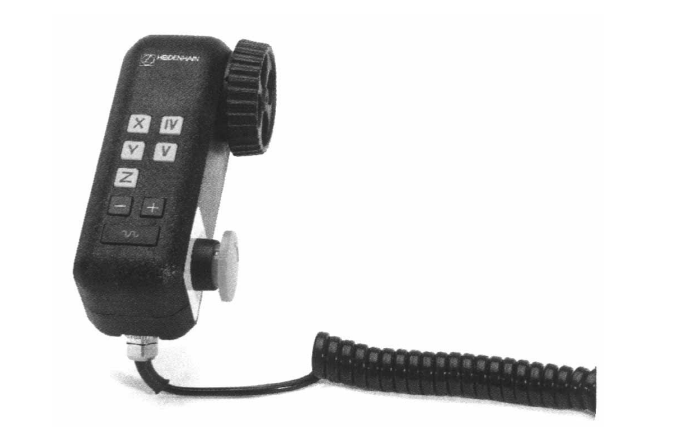


Fig. 1.7: The HR 330 Electronic Handwheel

1.2 Fundamentals of Numerical Control (NC)

Introduction

This chapter addresses the following topics:

- What is NC?
- The part program
- Programming
- Reference system
- Cartesian coordinate system
- Additional axes
- Polar coordinates
- Setting the pole
- Datum setting
- Absolute workpiece positions
- Incremental workpiece positions
- Programming tool movements
- Position encoders
- Reference mark evaluation

What is NC?

NC stands for Numerical Control. Simply put, numerical control is the operation of a machine by means of coded instructions. Modern controls such as the HEIDENHAIN TNCs have a built-in computer for this purpose. Such a control is therefore also called a CNC (Computer Numerical Control).

The part program

A part program is a complete list of instructions for machining a workpiece. It contains such information as the target position of a tool movement, the tool path — i.e. how the tool should move towards the target position — and the feed rate. The program must also contain information on the radius and length of the tools, the spindle speed and the tool axis.

Programming

The TNC is programmed in the ISO format; some programming sections, however, are guided by dialog prompting. The single commands (words) can be entered in any sequence within a block (except G90/G91). The TNC automatically sorts the single commands as soon as the block is concluded.

Reference system

In order to define positions one needs a reference system. For example, positions on the earth's surface can be defined "absolutely" by their geographic coordinates of longitude and latitude. The term "coordinate" comes from the Latin word for "that which is arranged", i.e. dimensions used for determining or defining positions. The network of horizontal and vertical lines around the globe constitutes an "absolute reference system" – in contrast to the "relative" definition of a position that is referenced, for example, to some other, known location.

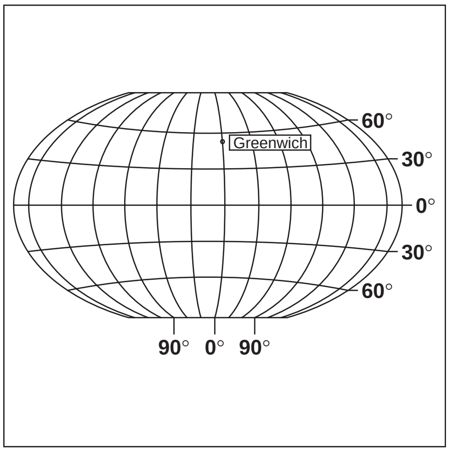


Fig. 1.8: The geographic coordinate system is an absolute reference system

Cartesian coordinate system

On a TNC controlled milling machine a workpiece is normally machined according to a workpiece-referenced Cartesian coordinate system (a rectangular coordinate system named after the French mathematician and philosopher René Descartes, Latin: Renatus Cartesius; 1596 to 1650). The Cartesian coordinate system is based on three coordinate axes X, Y and Z, which are parallel to the machine guideways. The figure to the right illustrates the "right hand rule" for remembering the three axis directions: the middle finger is pointing in the positive direction of the tool axis from the workpiece toward the tool (the Z axis), the thumb is pointing in the positive X direction, and the index finger in the positive Y direction.

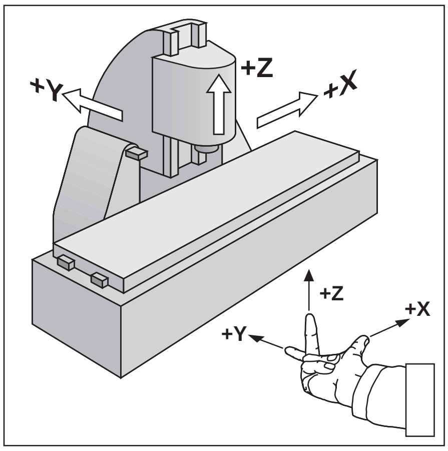


Fig. 1.9: Designations and directions of the axes on a milling machine

Additional axes

The TNC can control machines that have more than three axes. **U**, **V** and **W** are secondary linear axes parallel to the main axes X, Y and Z, respectively (see illustration). **Rotary axes** are also possible. They are designated as axes **A**, **B** and **C**.

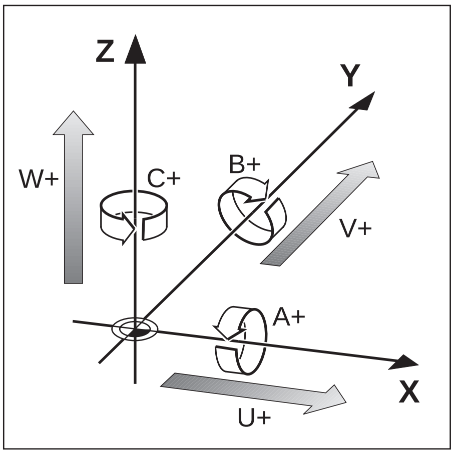


Fig. 1.10: Arrangement and designation of the auxiliary axes

Polar coordinates

The Cartesian coordinate system is especially useful for parts whose dimensions are mutually perpendicular. But when workpieces contain circular arcs, or when dimensions are given in degrees, it is often easier to use polar coordinates. In contrast to Cartesian coordinates, which are three-dimensional, polar coordinates can only describe positions in a plane.

The datum for polar coordinates is the **pole I, J, K**. To describe a position in polar coordinates, think of a scale whose zero point is rigidly connected to the pole but which can be freely rotated in a plane around the pole.

Positions in this plane are defined by:

- **Polar Radius R:** The distance from the pole I, J to the defined position.

- **Polar Angle H:** The angle between the reference axis and the scale.

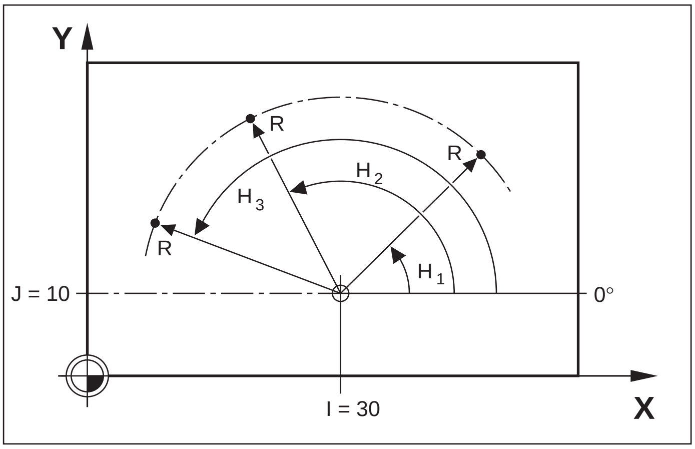


Fig. 1.11: Positions on an arc with polar coordinates

Setting the pole

The pole is defined by setting two Cartesian coordinates. These two coordinates also determine the reference axis for the polar angle PA.

Coordinates of the pole	Reference axis of the angle
I, J	+X
J, K	+Y
K, I	+Z

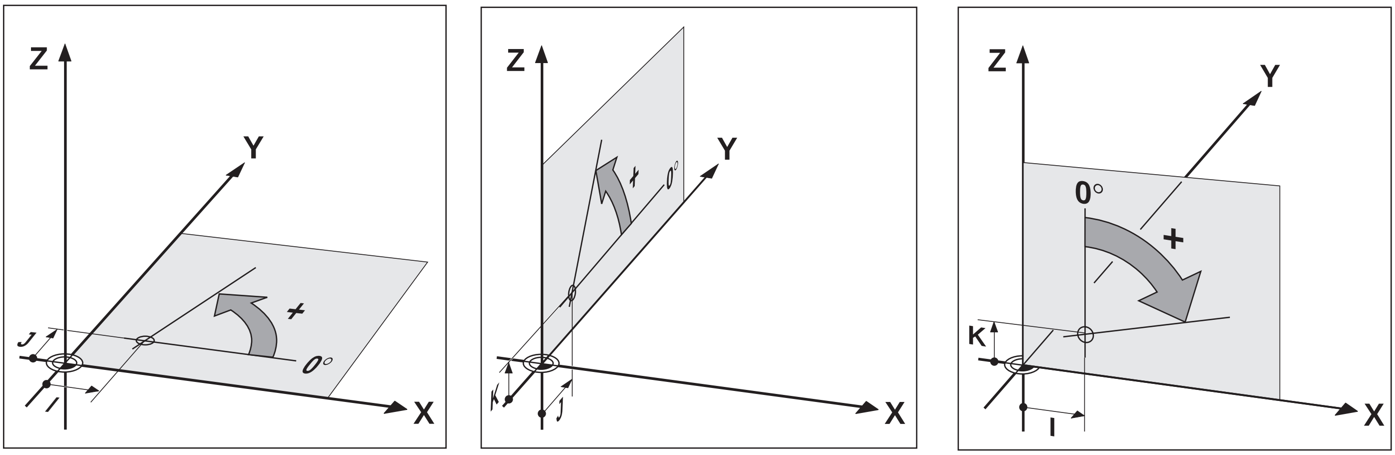


Fig. 1.12: Polar coordinates and their associated reference axes

Setting the datum

The workpiece drawing identifies a certain prominent point on the workpiece (usually a corner) as the "absolute datum" and perhaps one or more other points as relative datums. The process of datum setting establishes these points as the origin of the absolute or relative coordinate systems: The workpiece, which is aligned with the machine axes, is moved to a certain position relative to the tool and the display is set either to zero or to another appropriate position value (e.g. to compensate the tool radius).

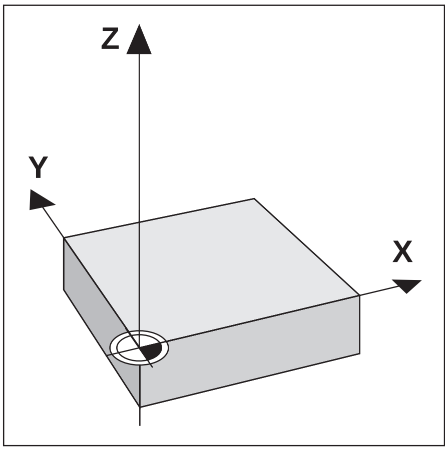
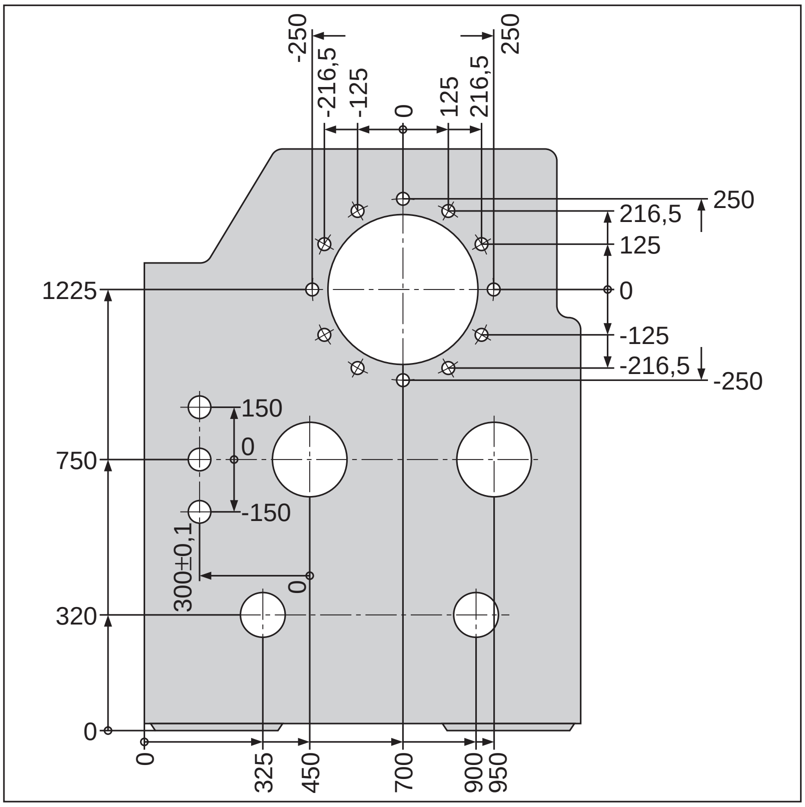


Fig. 1.13: The workpiece datum serves as the origin of the Cartesian coordinate system

Example:

**Drawings with several relative datums
(according to ISO 129 or DIN 406, Part 11; Figure 171)**



Example:

Coordinates of the point ①:

X = 10 mm
Y =   5 mm
Z =   0 mm

The datum of the Cartesian coordinate system is located 10 mm away from point ① on the X axis and 5 mm on the Y axis.

The 3D Touch Probe System from HEIDENHAIN is an especially convenient and efficient way to find and set datums.

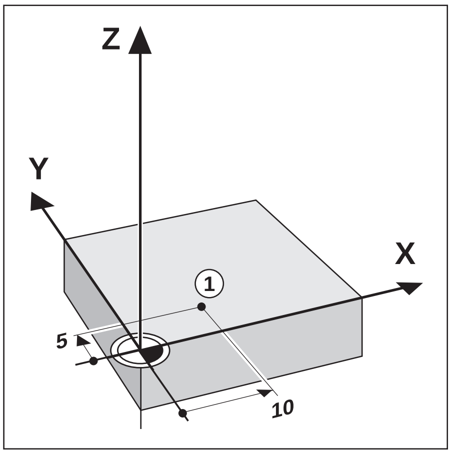


Fig. 1.15: Point ① defines the coordinate system.

Absolute workpiece positions

Each position on the workpiece is clearly defined by its absolute coordinates.

Example: Absolute coordinates of the position ①:
X = 20 mm
Y = 10 mm
Z = 15 mm

If you are drilling or milling a workpiece according to a workpiece drawing with absolute coordinates, you are moving the tool **to** the coordinates.

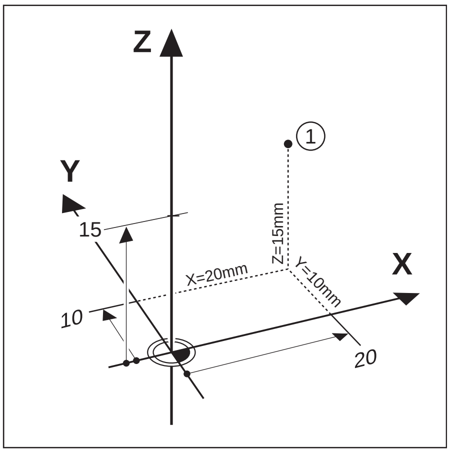


Fig. 1.16: Position ① of the example "absolute workpiece positions"

Incremental workpiece positions

A position can be referenced to the previous nominal position: i.e. the relative datum is always the last programmed position. Such coordinates are referred to as **incremental coordinates** (increment = growth), or also incremental or chain dimensions (since the positions are defined as a chain of dimensions). Incremental coordinates are designated with G91.

Example: Incremental coordinates of the position ③ referenced to position ②

Absolute coordinates of the position ② :
X = 10 mm
Y = 5 mm
Z = 20 mm

Incremental coordinates of the position ③ :
IX = 10 mm
IY = 10 mm
IZ = –15 mm

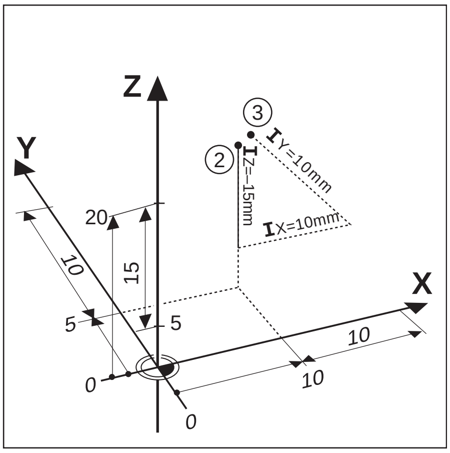


Fig. 1.17: Positions ② and ③ of the example "incremental workpiece positions"

If you are drilling or milling a workpiece according to a workpiece drawing with incremental coordinates, you are moving the tool **by** the coordinates.

An incremental position definition is therefore intended as an immediately relative definition. This is also the case when a position is defined by the **distance-to-go** to the target position (here the relative datum is located at the target position). The distance-to-go has a negative algebraic sign if the target position lies in the negative axis direction from the actual position.

The polar coordinate system can also express both types of dimensions:

- **Absolute polar coordinates** always refer to the pole I, J and the angle reference axis.
- **Incremental polar coordinates** always refer to the last programmed nominal position of the tool.

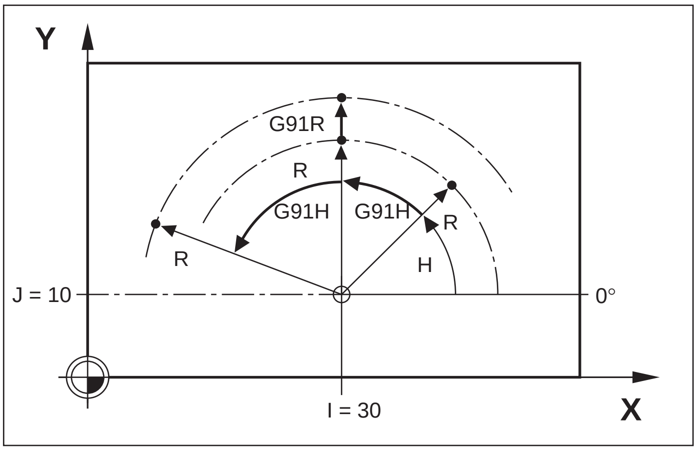
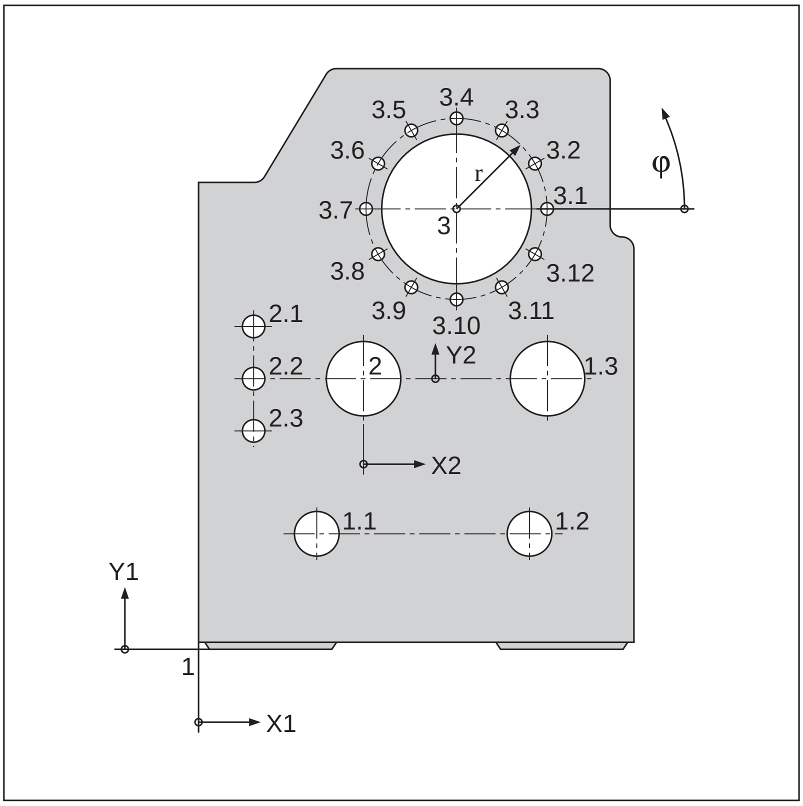


Fig. 1.18: Incremental dimensions in polar coordinates (designated with "G91")

Example:

Workpiece drawing with coordinate dimensioning (according to ISO 129 or DIN 406, Part 11; Figure 179)



		Dimensions in mm				
Coordinate origin		Coordinates				
	Pos.	X1 X2	Y1 Y2	r	φ	d
1	1	0	0			-
1	1.1	325	320			Ø 120 H7
1	1.2	900	320			Ø 120 H7
1	1.3	950	750			Ø 200 H7
1	2	450	750			Ø 200 H7
1	3	700	1225			Ø 400 H8
2	2.1	–300	150			Ø 50 H11
2	2.2	–300	0			Ø 50 H11
2	2.3	–300	–150			Ø 50 H11
3	3.1			250	0°	Ø 26
3	3.2			250	30°	Ø 26
3	3.3			250	60°	Ø 26
3	3.4			250	90°	Ø 26
3	3.5			250	120°	Ø 26
3	3.6			250	150°	Ø 26
3	3.7			250	180°	Ø 26
3	3.8			250	210°	Ø 26
3	3.9			250	240°	Ø 26
3	3.10			250	270°	Ø 26
3	3.11			250	300°	Ø 26
3	3.12			250	330°	Ø 26

Programming tool movements

An axis position is changed either by moving the tool or by moving the machine table on which the workpiece is fixed, depending on the individual machine tool.

> You always program as if the tool is moving and the workpiece is stationary.

If the machine table moves in one or several axes, the corresponding axes are designated on the machine operating panel with a prime mark (e.g. X', Y'). When an axis is designated with a prime mark, the programmed direction of axis movement is the opposite direction of tool movement relative to the workpiece.

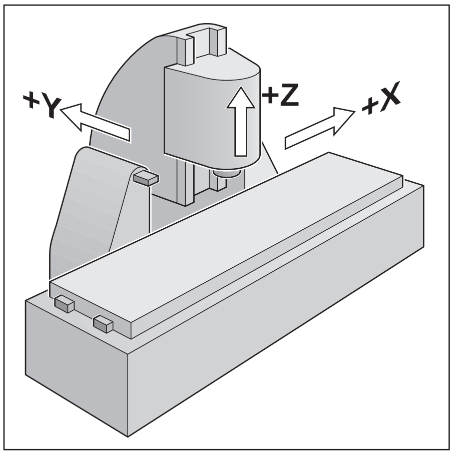


Fig. 1.20: On this machine the tool moves in the Y and Z axes; the machine table moves in the positive X' axis direction.

Position encoders

The position encoders – linear encoders for linear axes, angle encoders for rotary axes – convert the movement of the machine axes into electrical signals. The control evaluates these signals and constantly calculates the actual position of the machine axes.

If there is an interruption in power, the calculated position will no longer correspond to the actual position. When power is returned, the TNC can re-establish this relationship.

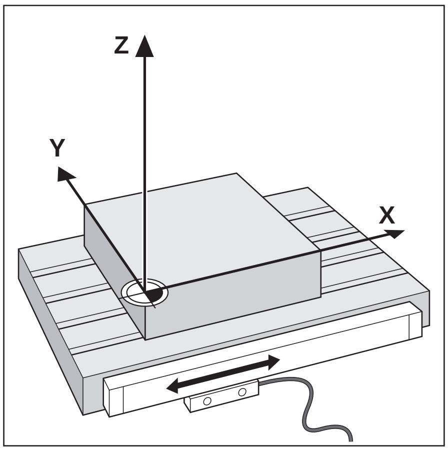


Fig. 1.21: Linear position encoder, here for the X axis

Reference marks

The scales of the position encoders contain one or more reference marks. When a reference mark is passed over, it generates a signal which identifies that position as the machine axis reference point.
With the aid of these reference marks the TNC can re-establish the assignment of displayed positions to machine axis positions.

If the position encoders feature **distance-coded** reference marks, each axis need only move a maximum of 20 mm (0.8 in.) for linear encoders, and 20° for angle encoders.

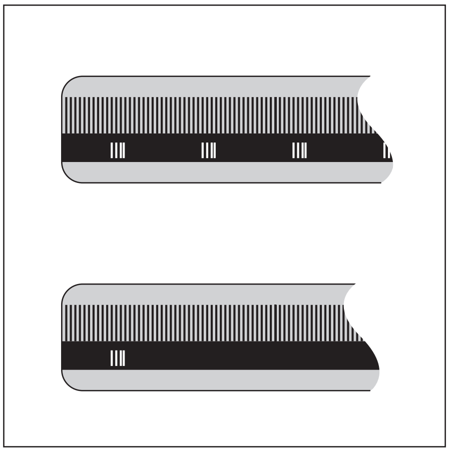


Fig. 1.22: Linear scales: above with distance-coded-reference marks, below with one reference mark

1.3 Switch-On

Switch on the power supply for the TNC and machine. The TNC then begins the following dialog:

MEMORY TEST

The TNC memory is automatically checked.

▼

POWER INTERRUPTED

Message from the TNC indicating that the power was interrupted.
Clear the message. ▶ CE

▼

TRANSLATE PLC PROGRAM

The PLC program of the TNC is automatically translated.

▼

RELAY EXT. DC VOLTAGE MISSING

Switch on the control voltage.
The TNC checks the functioning of the EMERGENCY STOP circuit. ▶ I

▼

MANUAL OPERATION

TRAVERSE REFERENCE POINTS

To cross over the reference marks in the displayed sequence:
Press the machine START button for each axis. ▶ I

To cross over the reference marks in any sequence:
For each axis, press and hold down the machine axis direction button until the reference mark has been crossed over. ▶ X, Y, ...

The TNC is now ready for operation
in the MANUAL OPERATION mode.

1.4 Graphics and Status Display

The TNC features various graphic display modes for testing programs. To be able to use this feature, you must select a program run operating mode.

Workpiece machining is simulated graphically in the display modes:

- Plan view
- Projection in three planes
- 3D view

With the fast internal image generation, the TNC calculates the contour and displays a graphic only of the completed part.

Select display mode

GRAPHICS 2 x MOD	Select display mode menu.
↑ / ↓	Select desired display mode.
ENT	Confirm selection.

Start graphic display

GRAPHICS START	Start graphic simulation in the selected display mode.

The START key repeats a graphic simulation as often as desired.

Rotary axis movements cannot be graphically simulated.
An attempted test run will result in an error message.

Plan view

In this mode, contour height is shown by image brightness.
The deeper the contour, the darker the image.

Number of depth levels: 7

This is the fastest of the three display modes.

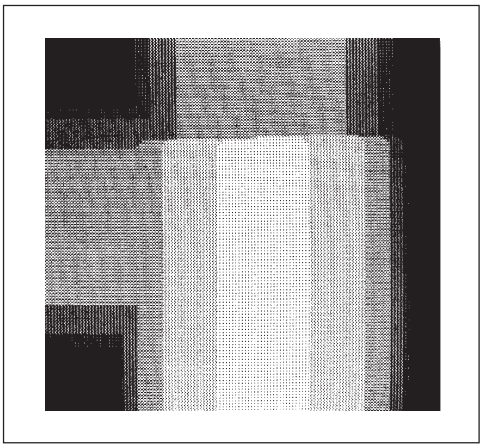


Fig. 1.23: TNC graphics, plan view

Projection in three planes

Here the program is displayed as in a technical drawing, with a plan view and two orthographic sections. A conical symbol near the graphic indicates whether the display is in first angle or second angle projection according to ISO 6433, Part 1. The type of projection can be selected with MP 7310.

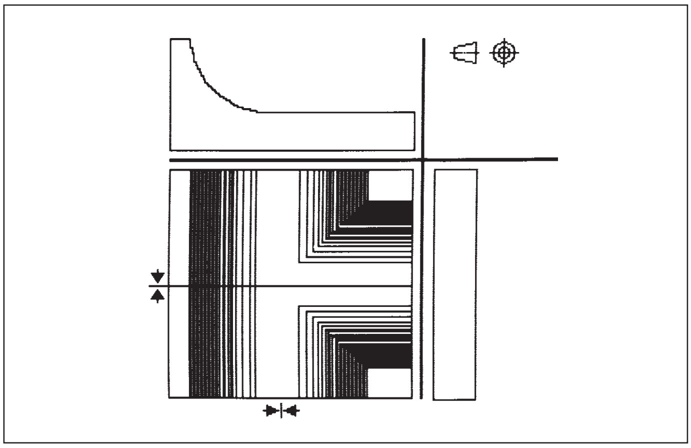


Fig. 1.24: TNC graphics, projection in three planes

Moving the sectional planes

The sectional planes can be moved to any position with the arrow keys. The position of the sectional plane is displayed on the screen while it is being moved.

3D view

This mode displays the simulated workpiece in three-dimensional space.

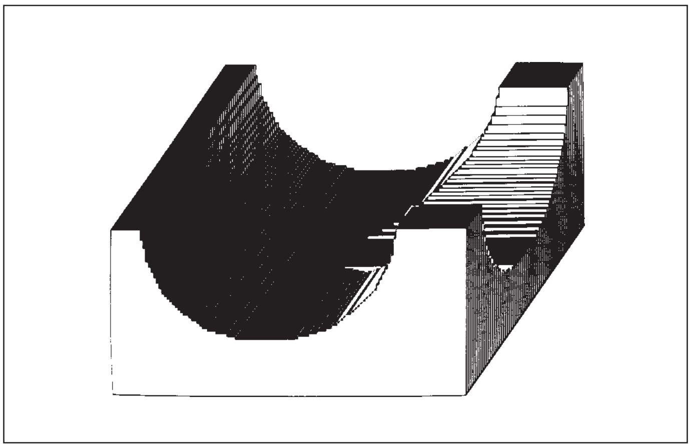


Fig. 1.25: TNC graphics, 3D view

Rotating the 3D view

In the 3D view, the image can be rotated around the vertical axis with the horizontal arrow keys. The angle of orientation is indicated with a special symbol:

└ 0^0 rotation

┘ 90^0 rotation

┐ 180^0 rotation

┌ 270^0 rotation

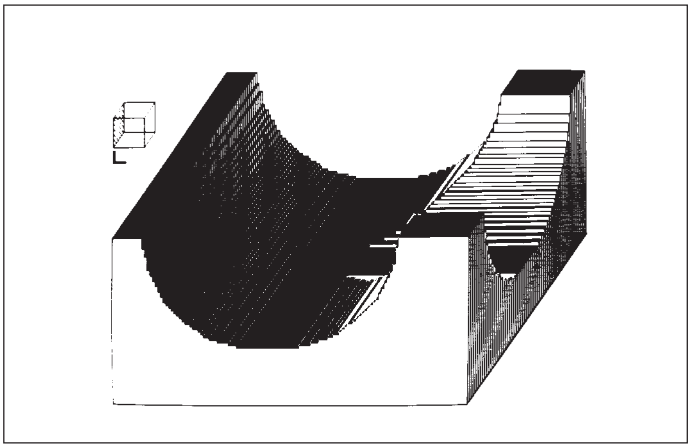


Fig. 1.26: Rotated 3D view

3D view, not true to scale

If the height-to-side ratio is between 0.5 and 50, a non-scaled 3D view can be selected with the vertical arrow keys. This view improves the resolution of the shorter workpiece side.

The angle orientation symbol also indicates the angle of orientation of the non-scaled 3D view.

Detail magnification of a 3D graphic

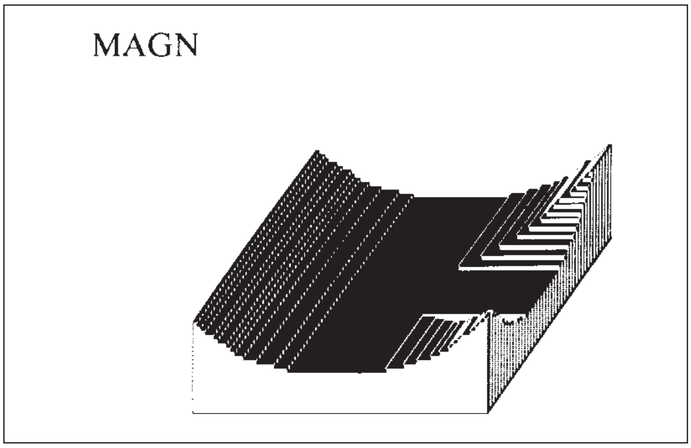


Fig. 1.27: Detail magnification of a 3D graphic

GRAPHICS MAGN	Select function for detail magnification.
↑ / ↓	Select sectional plane.
← / →	Set / reset section.
↓	If desired: switch dialog for transfer of detail.
TRANSFER DETAIL = ENT	
ENT	Magnify detail.

Details can be magnified in any display mode. The abbreviation MAGN appears on the screen to indicate that the image is magnified.

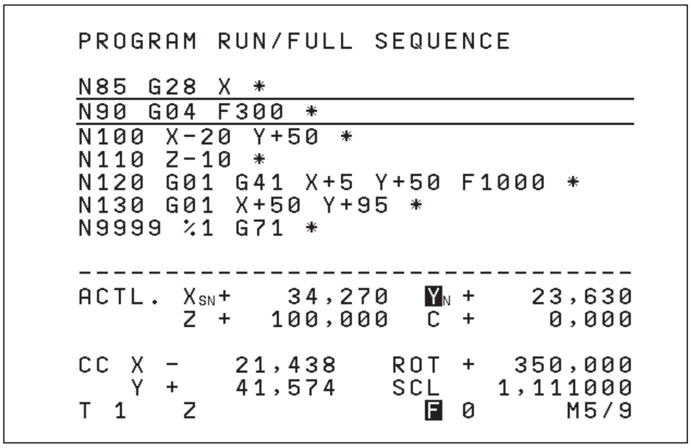
Return to non-magnified view

GRAPHICS BLK FORM	Press BLK FORM to display the workpiece in its programmed size.

Status Display

The status display in a program run operating mode shows the current coordinates as well as the following information:

- Type of position display (ACTL, NOML, ...)
- Axis locked (→|← in front of the axis)
- Number of current tool T
- Tool axis
- Spindle speed S
- Feed rate F
- Active miscellaneous function M
- TNC is in operation (indicated by ❇)
- Machines with gear ranges:
  Gear range following "/" character
  (depends on machine parameter)



```
PROGRAM RUN/FULL SEQUENCE

N85 G28 X *
N90 G04 F300 *
N100 X-20 Y+50 *
N110 Z-10 *
N120 G01 G41 X+5 Y+50 F1000 *
N130 G01 X+50 Y+95 *
N9999 %1 G71 *

--------------------------------
ACTL. XSN+   34,270  YN +   23,630
       Z +  100,000  C +     0,000

CC X -   21,438   ROT +  350,000
   Y +   41,574   SCL   1,111000
T 1   Z              F 0     M5/9
```

Fig. 1.28: Status display in a program run operating mode

> Bar graphs can be used to indicate analog quantities such as spindle speed and feed rate in the status display. These bar graphs must be activated by the machine tool builder.

1.5 Programs

The TNC 360 can store up to 32 part programs at once. The part programs can be written in HEIDENHAIN plain language dialog or according to ISO. ISO programs are indicated with “ISO”.

Each program is identified by a number with up to eight characters.

Program directory

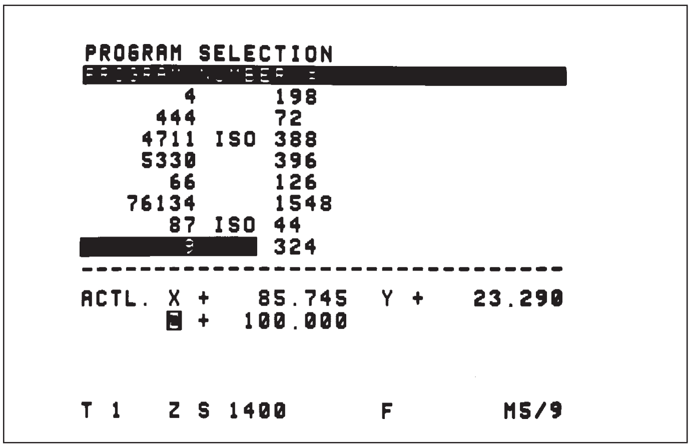
The program directory is called with the PGM NR key. To erase programs in the TNC memory, press the CL PGM key.

Action	Mode of operation	Call program directory with...
Create (a program)	⇨	... PGM NR
Edit	⇨	... PGM NR
Erase	⇨	... CL PGM
Test	⇨	... PGM NR
Execute	⇨ ⇨	... PGM NR

Fig. 1.29: Program management functions

The program directory provides the following information:

- Program number
- Program type (HEIDENHAIN or ISO)
- Program size in bytes, where one byte is the equivalent of one character.



```
PROGRAM SELECTION
PROGRAM NUMBER =
       4      198
     444       72
    4711 ISO  388
    5330      396
      66      126
   76134     1548
      87 ISO   44
       9      324
--------------------------------
ACTL. X +    85.745   Y +   23.290
      ■ +   100.000


T 1   Z S 1400      F       M5/9
```

Fig. 1.30: Program directory on the TNC screen

Selecting, erasing and protecting programs

To select a program:

PGM NR	Call the program management.

PROGRAM NUMBER ?

↑ or ↓	Use the arrow keys to highlight the program.
1 5	Enter the desired program number, for example 15.
ENT	Confirm your selection.

To erase a program:

CL PGM	Press CL PGM to call the program management.

ERASE = ENT / END = NO ENT

↑ or ↓	Use the arrow keys to highlight the program.
ENT or NO ENT	Erase the program or abort.

To protect a program:

PGM NR	Call the program management.

PROGRAM NUMBER = ?

e.g. 5 ENT	Enter the number of the program to be protected, for example program number 5.
→	Use the arrow key to highlight the first block.
G 5 0 END □	Enter the function for program protection, conclude the block.

*Resulting NC block: %5 G71 G50 **

Removing edit protection

To remove edit protection re-select the program and enter the code number 86357 with the corresponding MOD function (see page 10-3).

To remove edit protection:

Select the protected program, for example program number 5.

0 BEGIN 5 MM P	
MOD	Select MOD functions.

VACANT BYTES =	
↓ repeatedly	Activate the CODE NUMBER function.

CODE NUMBER	
8 6 3 5 7	Enter the code number 86357: Edit protection is removed, the "P" disappears.

2.1 Moving the Machine Axes

Traversing with the machine axis direction buttons

MANUAL OPERATION	
e.g. X	Press the machine axis direction button and hold it for as long as you wish the axis to move.

You can move several axes at once in this way.

For continuing movement:

MANUAL OPERATION	
e.g. Y I together	Press and hold the machine axis direction button, then press the machine START button. The axis continues to move after you release the keys.
O	To stop the axis, press the machine STOP button.

You can only move one axis at a time with this method.

Traversing with the electronic handwheel

ELECTRONIC HANDWHEEL

INTERPOLATION FACTOR: 1 3

e.g. 3 ENT	Enter the desired interpolation factor (see table below).
e.g. X	Select the axis that you wish to move: for portable handwheels, at the handwheel; for integral handwheels, at the TNC keyboard.

Now move the selected axis with the electronic handwheel. If you are using the portable handwheel, first press the enabling switch on its back.

Interpolation factor	Traverse in mm per revolution
0	20.000
1	10.000
2	5.000
3	2.500
4	1.250
5	0.625
6	0.312
7	0.156
8	0.078
9	0.039
10	0.019

Fig. 2.1: Interpolation factors and paths of traverse

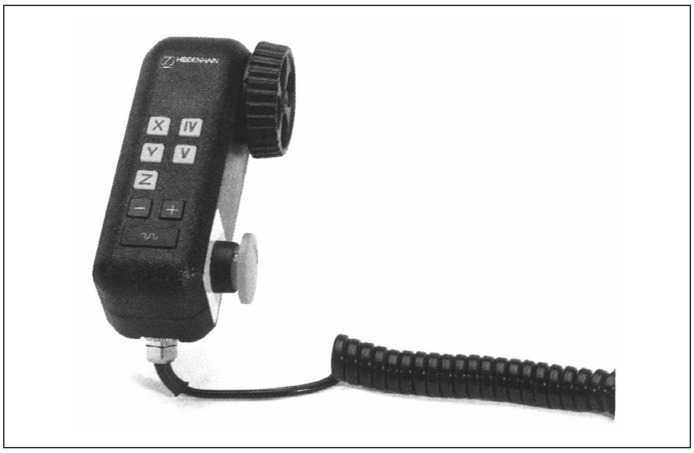


Fig. 2.2: HR 330 Electronic Handwheel

The smallest programmable interpolation factor depends on the individual machine tool.
Positioning with the electronic handwheel can also be carried out in the operating mode PROGRAMMING AND EDITING (depending on MP7641).

Working with the HR 330 Electronic Handwheel

Attach the electronic handwheel to a steel surface with the mounting magnets such that it cannot be operated unintentionally.

Be sure not to press the axis direction buttons unintentionally when you remove the handwheel from its position as long as the enabling switch (between the magnets) is depressed.

If you are using the handwheel for machine setup, press the enabling switch. Only then can you move the axes with the axis direction buttons.

Incremental jog positioning

With incremental jog positioning, a machine axis will move by a preset increment each time you press the corresponding machine axis direction button.

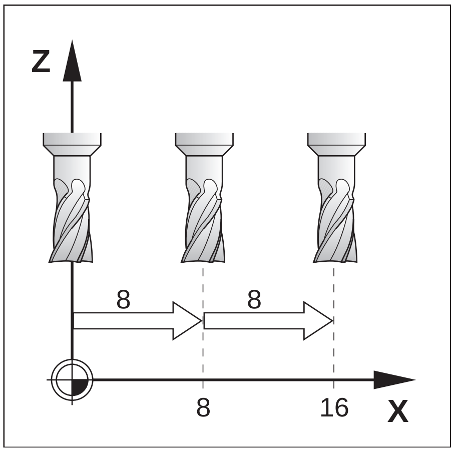


Fig. 2.3: Incremental jog positioning in the X axis

▶

ELECTRONIC HANDWHEEL

INTERPOLATION FACTOR: **4**

I	Select incremental jog positioning.
	Select incremental jog positioning by pressing the handwheel mode key again.

ELECTRONIC HANDWHEEL

JOG-INCREMENT: **4** **8**

e.g. 8 ENT	Enter the jog increment, for example 8 mm.
e.g. X	Press the machine axis direction button as often as desired.

Incremental jog positioning must be enabled by the machine tool builder.

Positioning with manual data input (MDI)

Page 5-41 describes positioning by manually entering the target coordinates for the tool.

2.2 Spindle Speed S, Feed Rate F and Miscellaneous Function M

The following values can be entered and changed in the MANUAL OPERATION and ELECTRONIC HANDWHEEL modes of operation:

- Miscellaneous function M
- Spindle speed S
- Feed rate F (can be changed but not entered)

For part programs these functions are entered or edited directly in the PROGRAMMING AND EDITING operating mode.

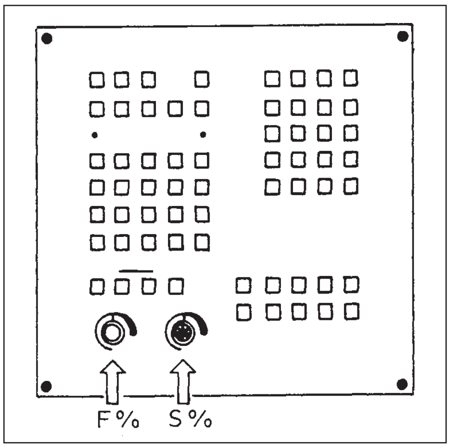


Fig. 2.4: Knobs for spindle speed and feed rate overrides

To enter the spindle speed S

S	Select the S function key.
N10 S	
e.g. 1 0 0 0 END □	Enter the spindle speed S, for example 1000 rpm.
I	Confirm the spindle speed S with the machine START button.

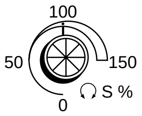
A miscellaneous function M starts spindle rotation at the entered speed S.

To enter the miscellaneous function M

M	Select the M function key.
N10 M	
e.g. 6 ENT	Enter the desired miscellaneous function M, for example M6.
I	Activate the miscellaneous function M with the machine START button.

Chapter 11 provides an overview of the miscellaneous functions.

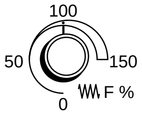
To change the spindle speed S

	Turn the spindle speed override knob: Adjust the spindle speed S to between 0% and 150% of the last entered value.

The spindle speed override will function only if your machine tool is equipped with a stepless spindle drive.

To change the feed rate F

In the MANUAL OPERATION mode the feed rate is set through a machine parameter.

	Turn the feed rate override knob: Adjust the feed rate to between 0% and 150% of the last entered value.

2.3 Setting the Datum without a 3D Touch Probe

You fix a datum by setting the TNC position display to the coordinates of a known point on the workpiece. The fastest, easiest and most accurate way of setting the datum is by using a 3D touch probe system from HEIDENHAIN (see page 2-14).

To prepare the TNC:

Clamp and align the workpiece.

Insert the zero tool with known radius into the spindle.

✋	Select the MANUAL OPERATION mode.

Ensure that the TNC is showing actual position values (see p. 10-4).

Setting the datum in the tool axis

Protective arrangement:
If the workpiece surface must not be scratched, you can lay a metal shim of known thickness d on it. Then enter a tool axis datum value that is larger than the desired datum by the value d.

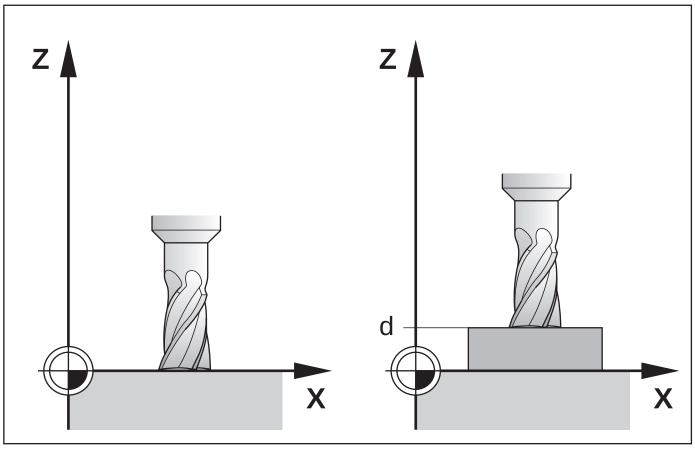


Fig. 2.5: Datum setting in the tool axis; right with protective shim

Move the tool until it touches workpiece surface.

e.g. Z	Select the tool axis.

DATUM SET Z =	
e.g. 0 ENT	For a zero tool: Set the display to Z = 0 or enter thickness d of the shim.
e.g. 5 0 ENT	For a preset tool: Set the display to the length L of the tool, for example Z=50 mm, or enter the sum Z=L+d.

Setting the datum in the working plane

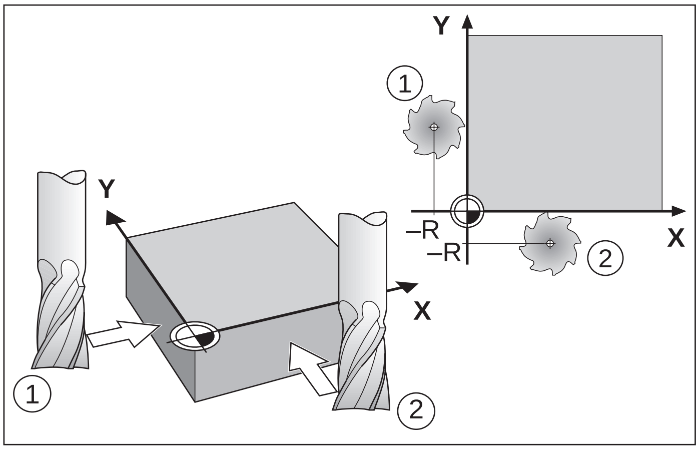


Fig. 2.6: Setting the datum in the working plane; plan view (upper right)

Move the zero tool until it touches the side of the workpiece.

e.g. X	Select the axis.
e.g. +/– 5 ENT	Enter the position of the tool center (here X = –5 mm) in the selected axis. Be careful to enter the correct algebraic sign.

Repeat the process for all axes in the working plane.

2.4 3D Touch Probe System

3D Touch probe applications

The TNC provides touch functions for application of a HEIDENHAIN 3D touch probe. Typical applications for the touch probe system are:

- Compensating workpiece misalignment (basic rotation)
- Datum setting
- Measuring:
  - Lengths and positions on the workpiece
  - Angles
  - Circle radii
  - Circle centers
- Measurements under program control
- Digitizing 3D surfaces (optional, only available with HEIDENHAIN plain language dialog programming.)

Fig. 2.7: HEIDENHAIN TS 120 three-dimensional touch probe system

The TNC must be specially prepared by the machine tool builder for the use of a 3D touch probe.

After you press the machine START button, the touch probe begins executing the selected probe function. The machine tool builder sets the feed rate F at which the probe approaches the workpiece. When the 3D touch probe contacts the workpiece, it

- transmits a signal to the TNC, which stores the coordinates of the probed position
- stops moving
- returns to its starting position in rapid traverse

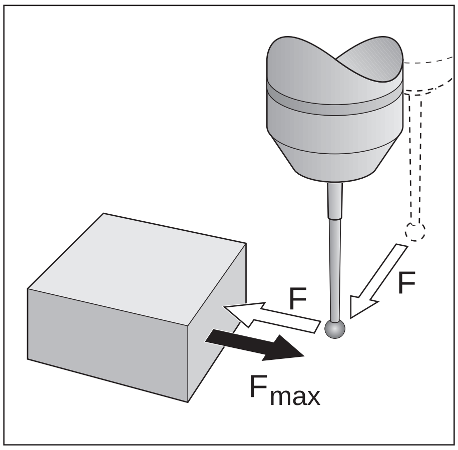


Fig. 2.8: Feed rates during probing

Selecting the touch probe menu

▶ **MANUAL OPERATION**

or

▶ **ELECTRONIC HANDWHEEL**

TOUCH PROBE	Select the menu of touch probe functions.

CALIBRATION EFFECTIVE LENGTH
CALIBRATION EFFECTIVE RADIUS
BASIC ROTATION
SURFACE = DATUM
CORNER = DATUM
CIRCLE CENTER = DATUM

Calibrating the 3D Touch Probe

The touch probe system must be calibrated

- for commissioning
- after a stylus breaks
- when the stylus is changed
- when the probe feed rate is changed
- in case of irregularities, such as those resulting from machine heating.

During calibration, the TNC finds the “effective” length of the stylus and the “effective” radius of the ball tip. To calibrate the 3D touch probe, clamp a ring gauge with known height and known internal radius to the machine table.

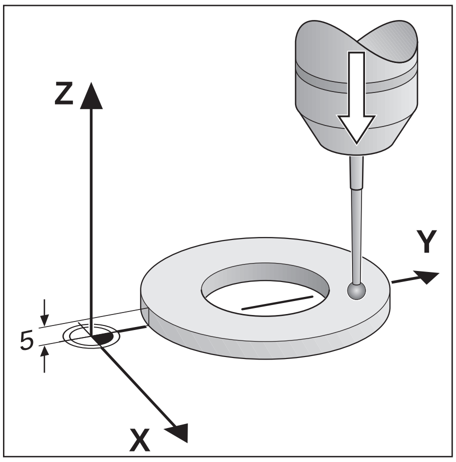


Fig. 2.9: Calibrating the touch probe length

To calibrate the effective length

Set the datum in the tool axis such that for the machine tool table, Z=0.

TOUCH PROBE ▶

SURFACE = DATUM

↓ / ↑ ENT	Select the calibration function for the touch probe length.

CALIBRATION EFFECTIVE LENGTH

Z+ Z–

TOOL AXIS = Z

e.g. Z	If necessary, enter the tool axis, for example Z.
↓	Move the highlight to DATUM.
e.g. 5	Enter the height of the ring gauge, for example 5 mm.

Move the touch probe to a position just above the ring gauge.

← or →	If necessary, change the displayed traverse direction.

I	The 3D touch probe contacts the upper surface of the ring gauge.

To calibrate the effective radius

Position the ball tip in the bore hole of the ring gauge.

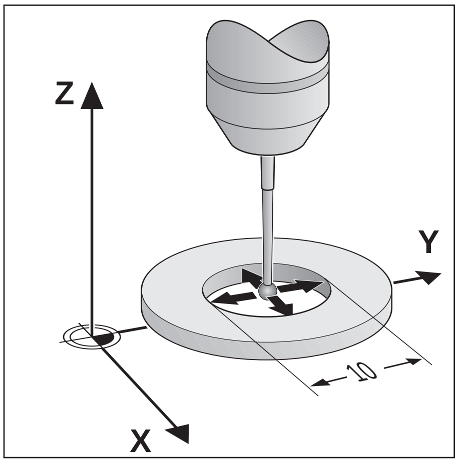


Fig. 2.10: Calibrating the touch probe radius

TOUCH PROBE ▶ **SURFACE = DATUM**

↓ / ↑ ENT	Select the calibration function for the ball-tip radius.

CALIBRATION EFFECTIVE RADIUS

X+ X– Y+ Y–

↓	Select RADIUS RING GAUGE.

RADIUS RING GAUGE = 0

5 ENT	Enter the radius of the ring gauge, for example 5 mm.
4 x I	The 3D touch probe contacts one position on the bore for each axis direction.

Displaying calibration values

The effective length and radius of the 3D touch probe are stored in the TNC for use whenever the touch probe is needed again. The stored values are displayed the next time the calibration function is called.

Compensating workpiece misalignment

The TNC electronically compensates workpiece misalignment by computing a “basic rotation.” Set the ROTATION ANGLE to the angle at which a workpiece surface should be oriented with respect to the angle reference axis (see p. 1-9) of the working plane.

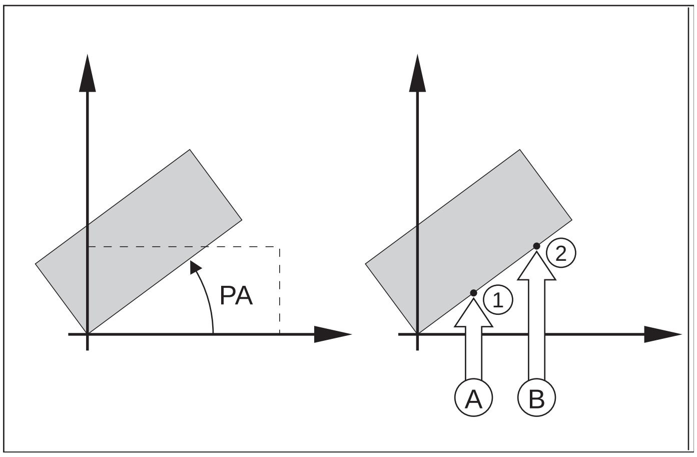


Fig. 2.11: Basic rotation of a workpiece, probing procedure for compensation (right). The dashed line is the nominal position; the angle PA is being compensated.

TOUCH PROBE ▶

SURFACE = DATUM

↓ / ↑ ENT	Select the BASIC ROTATION probe function.

BASIC ROTATION

X+ X- Y+ Y–

ROTATION ANGLE =

e.g. 0 ENT	Enter the nominal value of the ROTATION ANGLE.

Move the ball tip to a starting position Ⓐ near the first touch point ①.

X + X – Y + Y –

← or →	Select the probing direction.
I	Probe the workpiece.

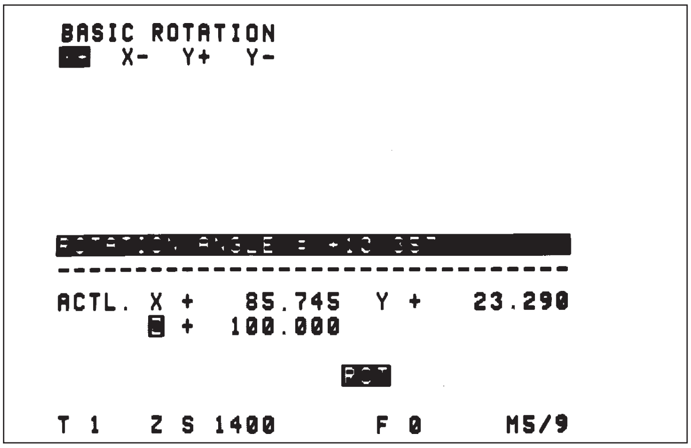
Move the ball tip to a starting position Ⓑ near the second touch point ②.

I	Probe the workpiece.

A basic rotation is kept in non-volatile storage and is effective for all subsequent program runs and graphic simulations.

Displaying basic rotation

The angle of the basic rotation is shown in the rotation angle display. When a basic rotation is active the abbreviation ROT is highlighted in the status display.



```
BASIC ROTATION
X+  X-   Y+   Y-




ROTATION ANGLE = +10.357
--------------------------------
ACTL. X +    85.745   Y +    23.290
      Z +   100.000

                  ROT

T 1   Z S 1400        F 0      M5/9
```

Fig. 2.12: Displaying the angle of an active basic rotation

To cancel a basic rotation:

Select BASIC ROTATION again.

ROTATION ANGLE =

0 ENT	Set the ROTATION ANGLE to 0.
END □	Terminate the probe function.

2.5 Setting the Datum with the 3D Touch Probe System

The following functions for setting the datum on an aligned workpiece are listed for in the TCH PROBE menu:

- Datum setting in any axis with SURFACE = DATUM
- Setting a corner as datum with CORNER = DATUM
- Setting the datum at a circle center with CIRCLE CENTER = DATUM

Setting the datum in a specific axis

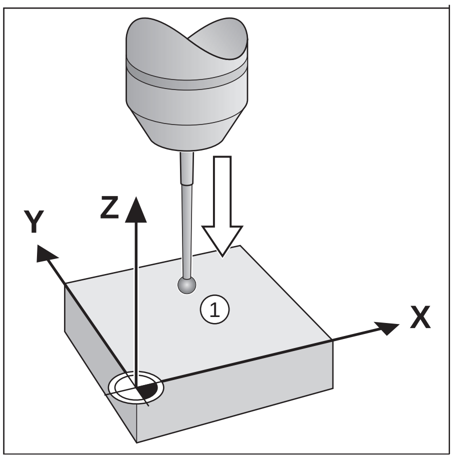


Fig. 2.13: Probing for the datum in the Z axis

Select the probe function SURFACE = DATUM.

Move the touch probe to a starting position near the touch point.

SURFACE = DATUM

X + X – Y + Y – Z + Z –

← or →	Select the probing direction and the axis in which you wish to set the datum, for example Z in the Z– direction.
I	Probe the workpiece.
e.g. 0 ENT	Enter the nominal coordinate of the DATUM.

Corner as datum

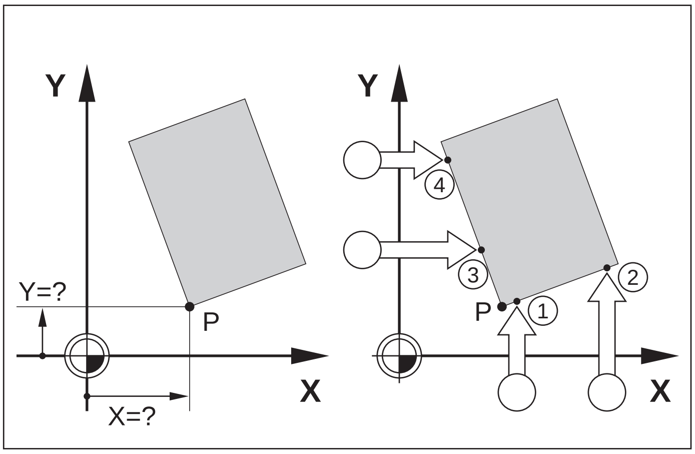


Fig. 2.14: Probing procedure for finding the coordinates of the corner P

Select the CORNER = DATUM probe function.

To use the points that were just probed for a basic rotation:

TOUCH POINTS OF BASIC ROTATION?

ENT	Transfer the touch point coordinates to memory.

Move the touch probe to a starting position near the first touch point on the side that was not probed for basic rotation.

CORNER = DATUM

X + X – Y + Y –

← or →	Select the probing direction.
I	Probe the workpiece.

Move the touch probe to a starting position near the second touch point on the same side.

I	Probe the workpiece.

DATUM X =

e.g. 0 ENT	Enter the first coordinate of the datum, for example in the X axis.

.
.
.

.
.
.

↓	Select the second coordinate.

DATUM Y =

e.g. 0 ENT	Enter the second coordinate of the datum, for example in the Y axis.

END □	Terminate the probe function.

If you do not wish to use points that were just probed for a basic rotation:

TOUCH POINTS OF BASIC ROTATION?

NO ENT	Ignore the dialog prompt.

Probe both workpiece sides twice.

Enter the datum coordinates.

Circle center as datum

With this function you can set the datum at the center of bore holes, circular pockets, cylinders, journals, circular islands etc.

Inside circle

The TNC automatically probes the inside wall in all four coordinate axis directions.

For incomplete circles (circular arcs) you can choose the appropriate probing direction.

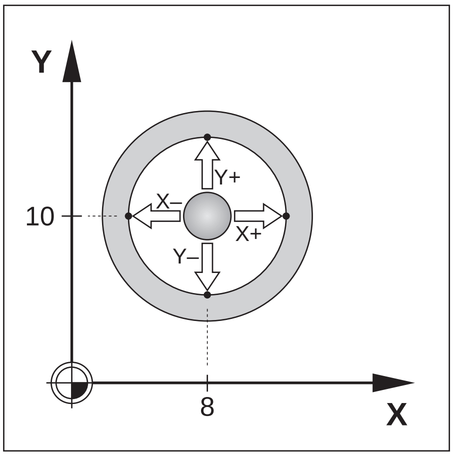


Fig. 2.15: Probing an inside cylindrical surface to find the center

Select the CIRCLE CENTER = DATUM probe function.

Move the touch probe to a position approximately in the center of the circle.

CIRCLE CENTER = DATUM

X + **X –** **Y +** **Y –**

4 x I	The probe touches four points on the inside of the circle.

DATUM X =

e.g. 8 ENT	Enter the first coordinate of the circle center, for example in the X axis.
↓	Select the second coordinate.

DATUM Y =

e.g. 1 0 ENT	Enter the second coordinate of the circle center, for example in the Y axis.
END □	Terminate the probe function.

Outside circle

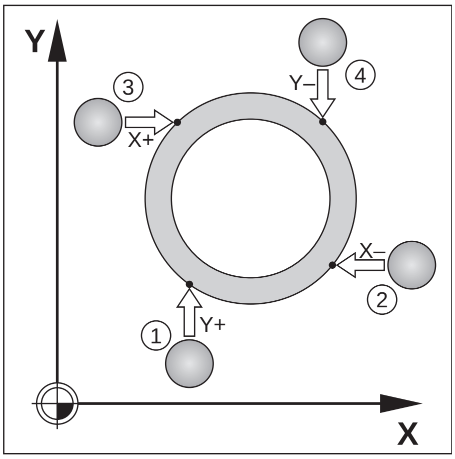


Fig. 2.16: Probing an outside cylindrical surface to find the center

Select the CIRCLE CENTER = DATUM probe function.

Move the touch probe to a starting position near the first touch point ① outside of the circle.

CIRCLE CENTER = DATUM

X + **X –** **Y +** **Y –**

← or →	Select the probing direction.
I	Probe the workpiece.

Repeat the probing process for points ②, ③ and ④ (see Fig. 2.16).

Enter the coordinates of the circle center.

After the probing procedure is completed, the TNC displays the coordinates of the circle center and the circle radius PR.

2.6 Measuring with the 3D Touch Probe System

With the 3D touch probe system you can determine

- Position coordinates, and from them,
- dimensions and angles on the workpiece.

Finding the coordinate of a position on an aligned workpiece

Select the SURFACE = DATUM probe function.

Move the touch probe to a starting position near the touch point.

SURFACE = DATUM

X + **X –** **Y +** **Y –** **Z +** **Z –**

← or →	Select the probing direction and the axis in which you wish to find the coordinate.
I	Probe the workpiece.

The TNC displays the coordinate of the touch point as DATUM.

Finding the coordinates of a corner in the working plane

Find the coordinates of the corner point as described under "Corner as datum." The TNC displays the coordinates of the probed corner as DATUM.

Measuring workpiece dimensions

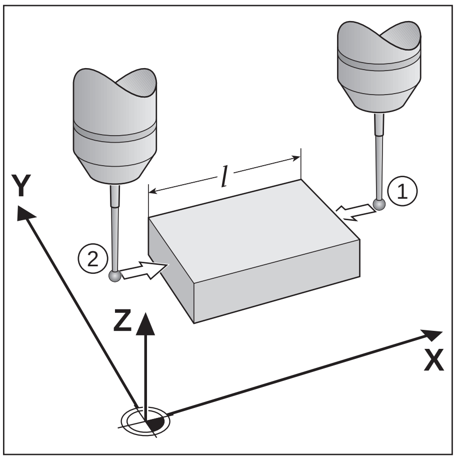


Fig. 2.17: Measuring lengths with the 3D touch probe

Select the SURFACE = DATUM probe function.

Move the probe to a starting position near the first touch point ①.

SURFACE = DATUM

X + X – Y + Y – Z + Z –

← or →	Use the arrow keys to select the probing direction.
I	Probe the workpiece.

If you will need the current datum later, write down the value that appears in the DATUM display.

DATUM X =

0 ENT	Set the DATUM to 0.
END □	Terminate the dialog.

Re-select the SURFACE = DATUM probe function.

Move the touch probe to a starting position near the second touch point ②.

.
.
.

⋮

SURFACE = DATUM

X + **X –** **Y +** **Y –** **Z +** **Z –**

← or →	Select the probing direction with the arrow keys – same axis as for ①.

I	Probe the workpiece.

The value displayed as DATUM is the distance between the two points on the coordinate axis.

To return to the datum that was active before the length measurement:

Select the SURFACE = DATUM probe function.

Probe the first touch point again.

Set the datum to the value that you wrote down previously.

END □	Terminate the dialog.

Measuring angles

You can also use the 3D touch probe system to measure angles in the working plane. You can measure

- the angle between the angle reference axis and a workpiece side, or
- the angle between two sides.

The measured angle is displayed as a value of maximum 90°.

To find the angle between the angle reference axis and a side of the workpiece:

Select the BASIC ROTATION probe function.

ROTATION ANGLE =

If you will need the current basic rotation later, write down the value that appears under ROTATION ANGLE.

Make a basic rotation with the side of the workpiece (see “Compensating workpiece misalignment”).

⋮

.
.
.

The angle between the angle reference axis and the side of the workpiece appears as the ROTATION ANGLE in the BASIC ROTATION function.

Cancel the basic rotation.

Restore the previous basic rotation by setting the ROTATION ANGLE to the value that you wrote down previously.

To measure the angle between two sides of a workpiece:

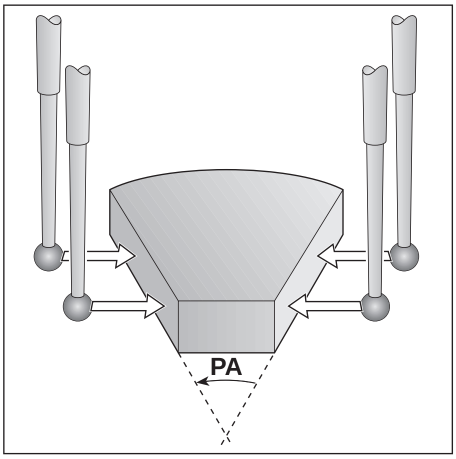


Fig. 2.18: Measuring the angle between two sides of a workpiece

Select the BASIC ROTATION probe function.

ROTATION ANGLE =

If you will need the current basic rotation later, write down the value that appears under ROTATION ANGLE.

Make a basic rotation for the first side (see “Compensating workpiece misalignment“).

Probe the second side as for a basic rotation, but do not set the ROTATION ANGLE to zero!

The angle PA between the workpiece sides appears as the ROTATION ANGLE in the BASIC ROTATION function.

Cancel the basic rotation.

Restore the previous basic rotation by setting the ROTATION ANGLE to the value that you wrote down previously.

3.1 Test Run

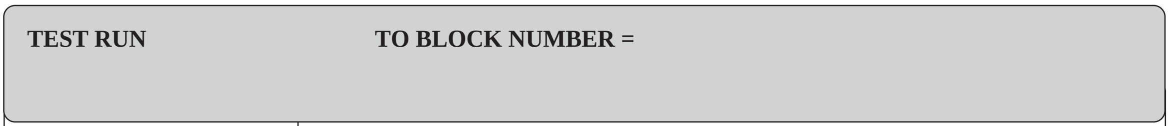
In the TEST RUN mode of operation the TNC checks programs and program sections for the following errors without moving the machine axes:

- Geometrical incompatibility
- Missing data
- Impossible jumps

The following TNC functions can be used in the TEST RUN operating mode:

- Test interruption at any block
- Optional block skip

To do a test run



TEST RUN TO BLOCK NUMBER =

NO ENT	Test the entire program.
e.g. 1 0 ENT	Test the program up to the entered block, for example block 10.

Test run functions

Function	Key
• Interrupt the test run	D
• Continue test run after interruption	NO ENT

3.2 Program Run

In the PROGRAM RUN / FULL SEQUENCE mode of operation the TNC executes a part program continuously to its end or up to a program stop.

In the PROGRAM RUN /SINGLE BLOCK mode of operation you execute each block separately by pressing the machine START button.

The following TNC functions can be used during a program run:

- Interrupt program run
- Start program run from a certain block
- Blockwise transfer of very long programs from external storage
- Checking/changing Q parameters
- Graphic simulation of a program run

To run a part program

- Clamp the workpiece to the machine table.
- Set the datum
- Select the program.

▶ **PROGRAM RUN / SINGLE BLOCK**

or

▶ **PROGRAM RUN / FULL SEQUENCE**

Select the part program.

GOTO □ 0 ENT	Go to the first block of the program.
I	Run the part program.
Only in mode PROGRAM RUN / SINGLE BLOCK I repeatedly	Run each block of the part program separately.

The feed rate and spindle speed can be changed with the override knobs.

Interrupting machining

There are various ways to interrupt a program run:

- Programmed interruptions
- External STOP key
- Switching to PROGRAM RUN / SINGLE BLOCK
- EMERGENCY STOP button

If the TNC registers an error during program run, it automatically interrupts machining.

Programmed interruptions

Interruptions can be programmed directly in the part program. The part program is interrupted at a block containing one of the following entries:

- G38
- Miscellaneous functions M0, M02 or M30
- Miscellaneous function M06, if the machine tool builder has assigned a stop function

To interrupt or abort machining immediately:

The block which the TNC is currently executing is not completed.

⊚	Interrupt machining.

The ❊ sign in the status display blinks.

The part program can be aborted with the D key.

D	Abort program run.

The ❊ sign disappears from the status display.

To interrupt machining by switching to the PROGRAM RUN / SINGLE BLOCK operating mode:

You can interrupt the program run at the end of the current block.

⊡	Select PROGRAM RUN / SINGLE BLOCK.

Resuming program run after an interruption

When a program run is interrupted the TNC stores:

- The data of the last called tool
- Active coordinate transformations
- The coordinates of the last defined circle center
- The count of a running program section repeat
- The number of the last block that calls a subprogram or a program section repeat

Resuming program run with the START button

You can resume program run by pressing the machine START button if the program was interrupted in one of the following ways:

- Pressing the machine STOP button
- A programmed interruption
- Pressing the EMERGENCY STOP button (machine-dependent function).

Resuming program run after an error

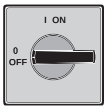
- If the error message is not blinking:

Remove the cause of the error.

CE	Clear the error message from the screen.

Restart the program.

- If the error message is blinking:

	Switch off the TNC and the machine.

Remove the cause of the error.

Restart the program.

- If you cannot correct the error:

Write down the error message and contact your repair service agency.

3.3 Blockwise Transfer: Executing Long Programs

Part programs that occupy more memory than the TNC provides can be "drip fed" block by block from an external storage device.

During program run, the TNC transfers program blocks from a floppy disk unit or PC through its data interface, and erases them after execution.

To prepare for blockwise transfer:

- Prepare the data interface.
- Configure the data interface with the MOD function (see page 10-3).
- If you wish to transfer a part program from a PC, adapt the TNC and PC to each other (see pages 9-4 and 11-2).
- Ensure that the transferred program meets the following requirements:
  - The highest block number must not exceed 65534. However, the block numbers can repeat themselves as often as necessary.
  - All programs called from the transferred program must be present in the TNC memory
  - The transferred program must not contain:
    Subprograms
    Program section repetitions
    The function D 15:PRINT
  - The TNC can store up to 20 G99 blocks.

PROGRAM RUN / SINGLE BLOCK

or

TEST RUN

EXT	Select the function for blockwise transfer.

PROGRAM NUMBER

e.g. 1 0 ENT	Enter the program number and start data transfer.
I	Execute the transferred program blocks.

If the data transfer is interrupted, press the machine START button again.

Jumping over blocks

The TNC can jump to any desired block in the program to begin transfer.
The preceding blocks are ignored during a program run.

Select the program and start transfer.

GOTO □ e.g. 1 5 0 ENT	Enter the block number at which you wish to begin data transfer, for example 150.
I	Execute the transferred blocks, starting with the block number that you entered.

4 Programming

In the PROGRAMMING AND EDITING mode of operation you can do such things as

- creating,
- adding to, and
- editing files.

This chapter describes basic functions and programming input that do not cause machine axis movement. The entry of geometry for workpiece machining is described in the next chapter.

4.1 Editing Part Programs

Layout of a program

A part program consists of individual program blocks.

The TNC numbers the blocks in ascending order. The block number increment is defined through the machine parameter MP 7220 (see page 11-5). Program blocks contain units of information called "words".

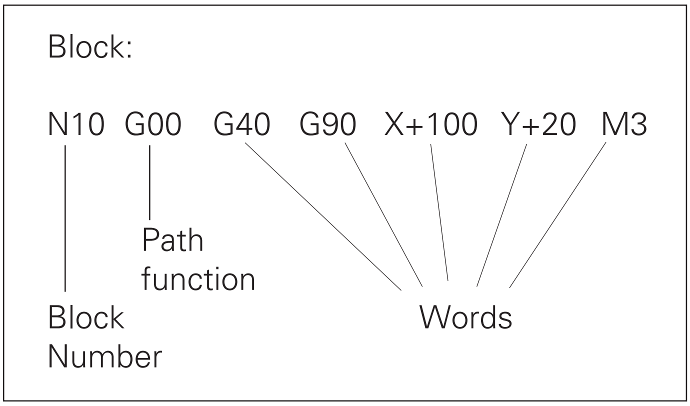


Fig. 4.1: Program blocks contain words of specific information

Function	Key
• Continue the dialog	ENT
• Ignore the dialog question	NO ENT
• End the block	END □
• Erase the block / Erase the word	DEL □

Editing functions

Editing means entering, adding to or changing commands and information for the TNC.

The TNC enables you to

- Enter data with the keyboard
- Select desired blocks and words
- Insert and erase blocks and words
- Correct erroneously entered values and commands
- Easily clear TNC messages from the screen

Types of input

Numbers, coordinate axes and radius compensation are entered directly by keyboard. You can set the algebraic sign either before, during or after a numerical entry.

Selecting blocks and words

- To call a block with a certain block number:

GOTO □ e.g. 1 0 ENT	Block number 10 is highlighted.

- To move one block forward or backward:

↑ or ↓	Press the vertical arrow keys.

- To select individual words in a block:

← or →	Press the horizontal arrow keys.

- To find the same word in other blocks:

← or →	Select the word in the block.
↑ or ↓	Jump to the same word in other blocks.

Inserting blocks

Additional program blocks can be inserted behind any existing block (except the N9999 block).

↑ or ↓ / GOTO □	Select the block in front of the desired insertion.
N e.g. 3 5 ENT	Program the new block.

Editing and inserting words

Highlighted words can be changed as desired: simply overwrite the old value with the new one. After entering the new information, press a horizontal arrow key to remove the highlight from the block or confirm the change with the END key. You can also insert new words into a specific block by moving the highlight to the desired block with the horizontal arrow keys.

Erasing blocks and words

Function	Key
• Set the selected number to 0	CE
• Erase an incorrect number	CE
• Clear a non-blinking error message	CE
• Delete the selected word	DEL □
• Delete the selected block	DEL □
• Erase program sections: First select the last block of the program section to be erased.	DEL □

4.2 Tools

Each tool is identified by a number.

The tool data, consisting of the:

- Length L, and
- Radius R

are assigned to the tool number.

The tool data can be entered:

- into the individual part program in a G99 block, or
- once for each tool into a common tool table that is stored as program 0.

Once a tool is defined, the TNC then associates its dimensions with the tool number and accounts for them when executing positioning blocks.

Determining tool data

Tool number

Each tool is designated with a number between 0 and 254.

The tool with the number 0 is defined as having length L = 0 and radius R = 0. In tool tables, T0 should also be defined with L = 0 and R = 0.

Tool radius R

The radius of the tool is entered directly.

Tool length L

The compensation value for the tool length is measured

- as the difference in length between the tool and a zero tool, or
- with a tool pre-setter.

A tool pre-setter eliminates the need to define a tool in terms of the difference between its length and that of another tool.

Determining tool length with a zero tool

For the sign of the tool length L:

$L > L_0$ A positive value means the tool is longer than the zero tool.

$L < L_0$ A negative value means the tool is shorter than the zero tool.

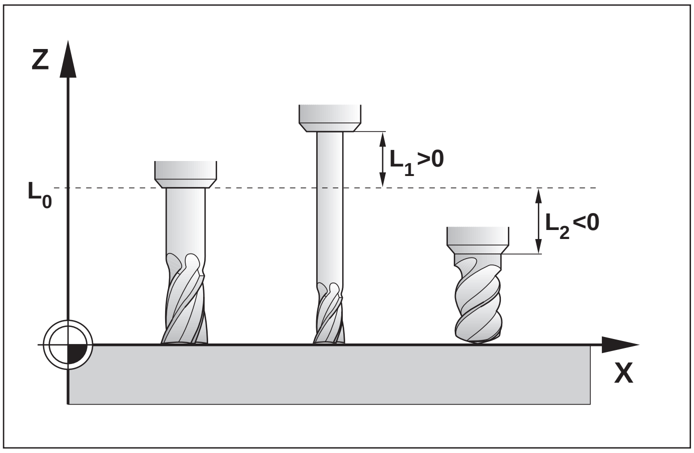


Fig. 4.2: Tool lengths can be given as the difference from the zero tool

Move the zero tool to the reference position in the tool axis (e.g. workpiece surface with Z = 0).

If necessary, set the datum in the tool axis to 0.

Change tools.

Move the new tool to the same reference position as the zero tool.

The TNC displays the compensation value for the length L of the tool.

Write the value down and enter it later.

Enter the display value by using the "actual position capture" function (see page 4-20).

Entering tool data into the program

The following data can be entered for each tool in the part program:

- Tool number
- Tool length compensation value L
- Tool radius R

To enter tool data in the program block:

G 9 9 ENT ▶

▶ TOOL NUMBER T	
e.g. 5 ENT	Designate the tool with a number, for example 5.

TOOL LENGTH L	
1 0 ENT	Enter the compensation value for the tool length, for example L = 10mm.

TOOL RADIUS R	
e.g. 5 END □	Enter the tool radius, for example R = 5mm.

Resulting NC block: G99 T5 L+10 R+5

You can enter the tool length L directly in the tool definition by using the "actual position capture" function (see page 4-20).

Entering tool data in program 0

The data for all tools can be entered in a common tool table. The number of tools in the table is selected through the machine parameter MP 7260.

If your machine uses an automatic tool changer, the tool data must be stored in the tool table.

Editing the tool table (program 0)

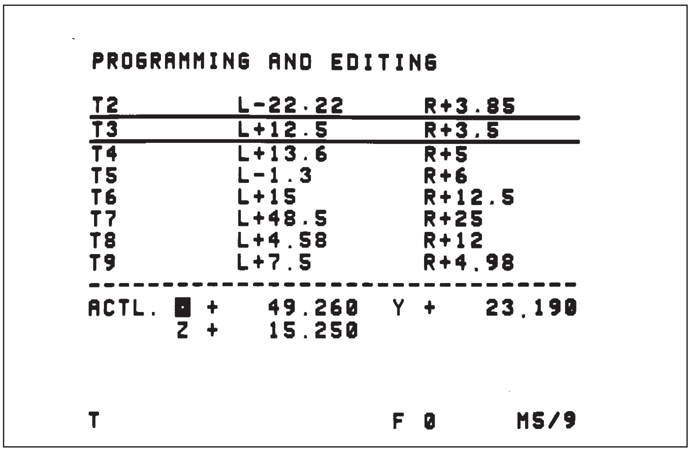
PROGRAMMING AND EDITING	
PGM NR	Call the program directory.

PROGRAM NUMBER =	
0 ENT	Select the tool table.

In the ELECTRONIC HANDWHEEL and MANUAL modes of operation, you can call the tool table at any time by simply pressing ENT.

Data in the tool table

The tool table contains further information in addition to the tool dimensions.



```
PROGRAMMING AND EDITING

T2         L-22.22      R+3.85
T3         L+12.5       R+3.5
T4         L+13.6       R+5
T5         L-1.3        R+6
T6         L+15         R+12.5
T7         L+48.5       R+25
T8         L+4.58       R+12
T9         L+7.5        R+4.98
---------------------------------
ACTL. ■ +   49.260  Y +   23.190
      Z +   15.250


T                   F 0     M5/9
```

Fig. 4.3: Tool table

Abbreviation	Input	Dialog
T	**T**ool number: the number with which the tool is called in a part program	–
S	**S**pecial tool with large radius requiring more than one pocket in the tool magazine. A certain number of pockets is kept vacant on each side of the special tool. The letter S then appears in front of the tool number.	SPECIAL TOOL? YES = ENT / NO = NO ENT
P	**P**ocket number of the tool in the magazine	POCKET NUMBER?
L	Compensation value for the **L**ength of the tool	TOOL LENGTH L?
R	**R**adius of the tool	TOOL RADIUS R?

Calling tool data

The following data can be programmed in the T block:

- Tool number, Q parameter
- Working plane with G17/G18 or G19
- Spindle speed S

To call the tool data:

T ▶ **TOOL NUMBER?**

e.g. 5	Enter the number of the tool as it was defined in the tool table or in a "G99" block, for example 5.
G 1 7	Select the spindle axis Z.
S 5 0 0 END □	Enter the desired spindle speed, for example S = 500 rpm.

Resulting NC block: T5 G17 S500

Tool pre-selection with tool tables

If you are using tool tables, you can indicate which tool you will next need by entering a G51 block. Simply enter the tool number or a corresponding Q parameter.

Tool change

Automatic tool change

If your machine is built for automatic tool changing, the TNC controls the replacement of the inserted tool by another from the tool magazine. The program run is not interrupted.

Manual tool change

To change the tool manually, stop the spindle and move the tool to the tool change position. Sequence of action:

- Move to the tool change position (under program control, if desired)
- Interrupt program run (see page 3-4)
- Change the tool
- Continue the program run (see page 3-5)

Tool change position

A tool change position must lie next to or above the workpiece to prevent tool collision. With the miscellaneous functions M91 and M92 (see page 5-39) you can enter machine-referenced rather than workpiece-referenced coordinates for the tool change position.

If T0 is programmed before the first tool call, the TNC moves the spindle to an uncompensated position.

If a positive length compensation value was in effect before T0, the clearance to the workpiece is reduced.

4.3 Tool Compensation Values

For each tool, the TNC adjusts the spindle path in the tool axis by the compensation value for the tool length. In the working plane it compensates the tool radius.

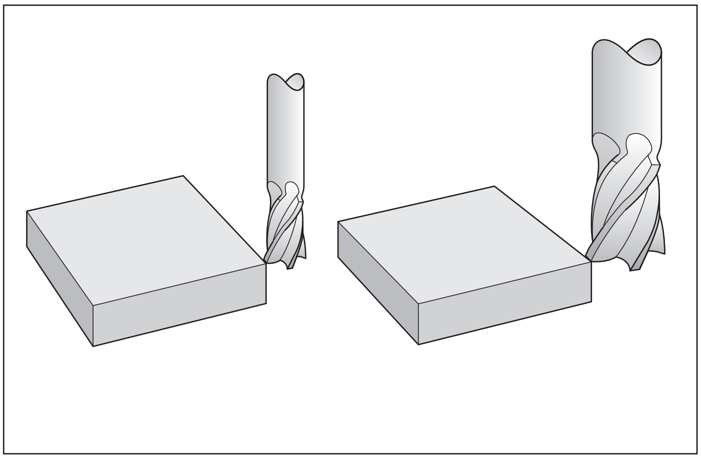


Fig. 4.4: The TNC must compensate the length and radius of the tool

Effect of tool compensation values

Tool length

Length compensation becomes effective automatically as soon as a tool is called and the tool axis moves.

To cancel length compensation, call a tool with the length L = 0.

Tool radius

Radius compensation becomes effective as soon as a tool is called and is moved in the working plane with G41 or G42.

To cancel radius compensation, program a positioning block with G40.

Tool radius compensation

Tool traverse can be programmed:

- Without radius compensation: G40
- With radius compensation: G41or G42
- As single-axis movements with G43 or G44

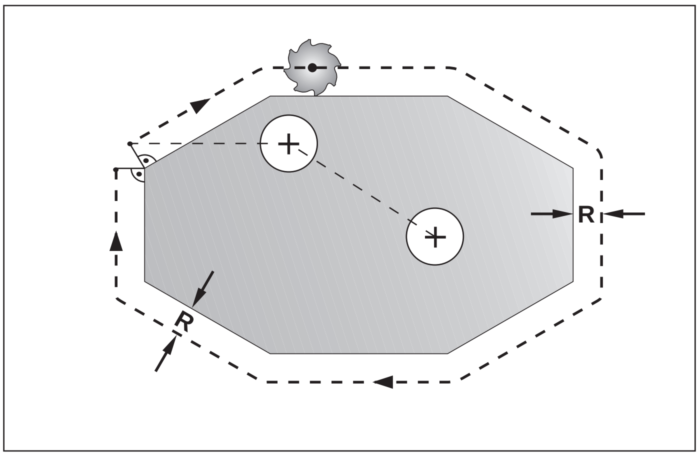


Fig. 4.5: Programmed contour (——, +) and the path of the tool center (- - -)

Traverse without radius compensation: G40

The tool center moves to the programmed coordinates.

Applications:

- Drilling and boring
- Pre-positioning

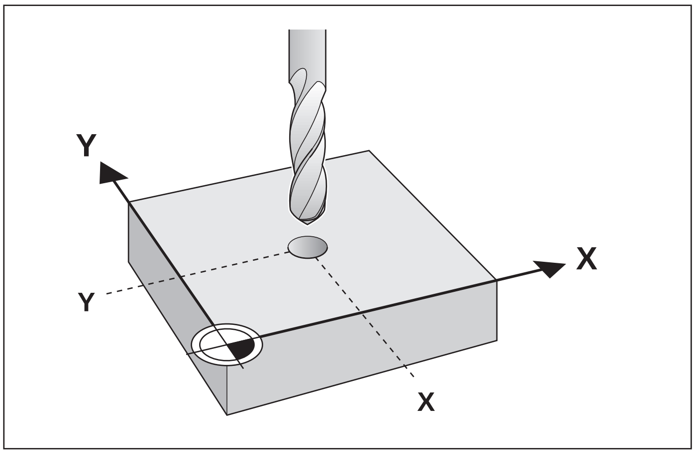


Fig. 4.6: These drilling positions are entered without radius compensation

Traverse with radius compensation G41, G42

The tool center moves to the left (G41) or to the right (G42) of the programmed contour at a distance equal to the tool radius. "Right" or "left" is meant as seen in the direction of tool movement as if the workpiece were stationary.

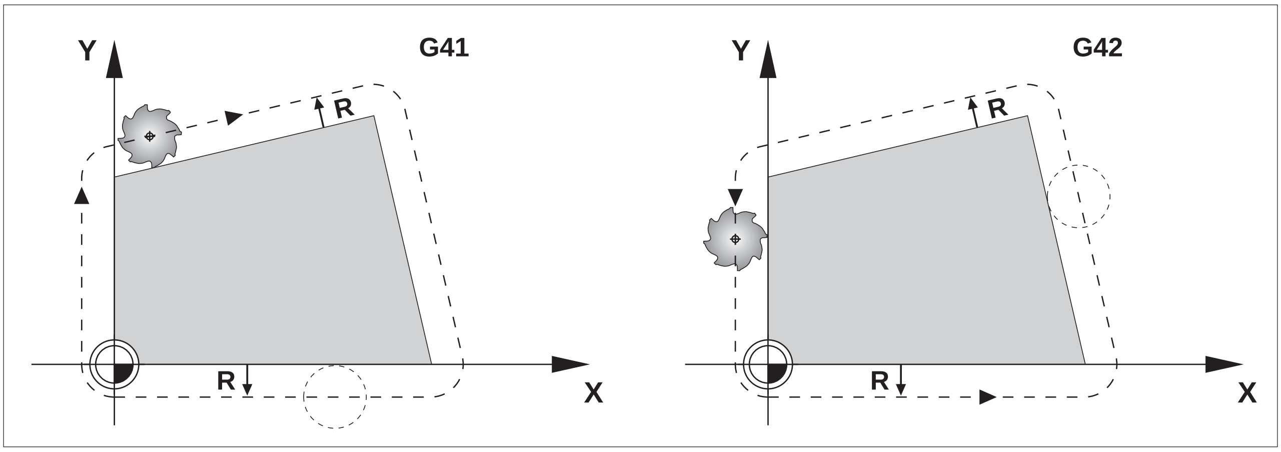


Fig. 4.7: The tool moves to the left (G41) or to the right (G42) of the workpiece during milling

Between two program blocks with differing radius compensation you must program at least one block without radius compensation (that is, with G40). Radius compensation is not in effect until the end of the block in which it is first programmed.

Shortening or lengthening single-axis movements G43, G44

This type of radius compensation is possible only for single-axis movements in the working plane: The programmed tool path is shortened (G44) or lengthened (G43) by the tool radius.

Applications:

- Single-axis machining
- Occasionally for pre-positioning the tool, such as for cycle G47: SLOT MILLING.

- G43 and G44 are activated by programming a positioning block with only one axis.
- The machine tool builder may block the entry of single-axis positioning blocks through a machine parameter.

Machining corners

Outside corners

The TNC moves the tool in a transitional arc around outside corners. The tool "rolls around" the corner point.

If necessary, the feed rate F is automatically reduced at outside corners to reduce machine strain, for example for very sharp changes in direction.

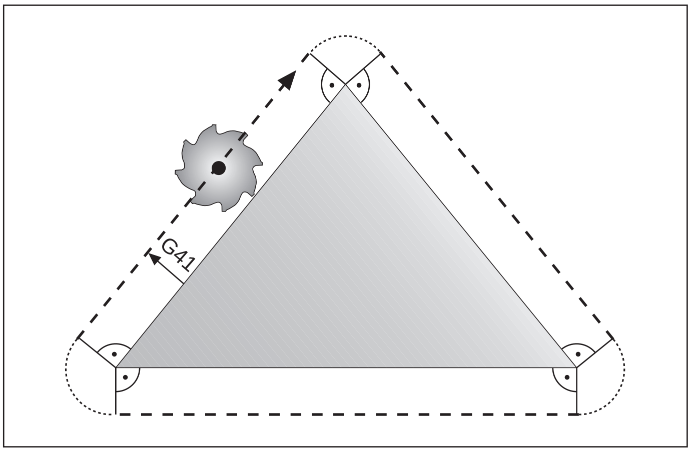


Fig. 4.8: The tool "rolls around" outside corners

If you work without radius compensation, you can influence the machining of outside corners with the miscellaneous function M90 (see page 5-36).

Inside corners

The TNC calculates the intersection of the tool center paths at inside corners. From this point it then starts the next contour element. This prevents damage to the workpiece at inside corners.

When two or more inside corners adjoin, the chosen tool radius must be small enough to fit in the programmed contour.

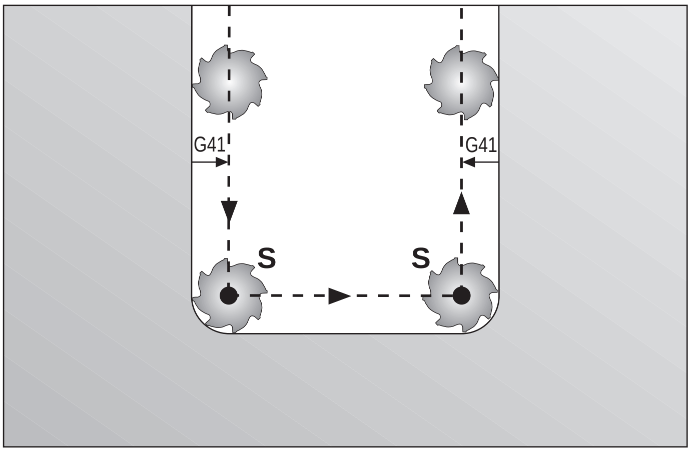


Fig. 4.9: Tool path for inside corners

4.4 Program Creation

To create a new part program

PGM NR	Call the file directory.

Select any program.

PROGRAM NUMBER=

e.g. 7 4 3 ENT	Enter the name of the new program, for example 743.

MM=G71 / INCH=G70

e.g. 7 1 ENT	Select the unit of measurement used in the program, for example millimeters (G71), conclude the block.

Defining the blank form

If you wish to use the graphic workpiece simulation you must first define a rectangular workpiece blank. Its sides lie parallel to the X, Y and Z axes and can be up to 30 000 mm long.

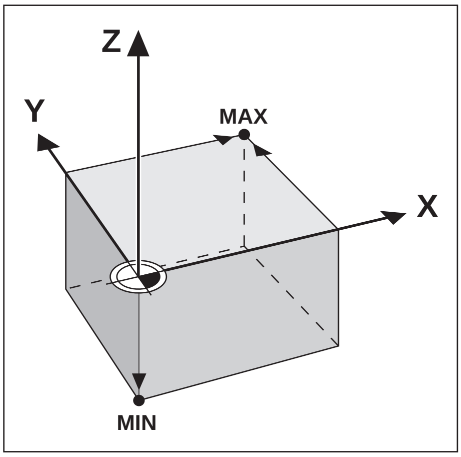


Fig. 4.10: The MIN and MAX points define the blank form

The ratio of the blank-form side lengths must be less than 84:1.

MIN and MAX points

The blank form is defined by two of its corner points:

- The MIN point — the smallest X, Y and Z coordinates of the blank form, entered as absolute values.
- The MAX point — the largest X, Y and Z coordinates of the blank form, entered as absolute values or incremental values.

G 3 0	G function for entering the MIN point.
G 1 7	Select the tool axis: G17 designates the Z axis.
e.g. X 0 Y 0 Z 4 0 +/– END □	Enter the MIN point coordinates for the X, Y and Z axes; confirm the block with END.

G 3 1	G function for entering the MAX point.
G 9 0	Enter an absolute value, or
G 9 1	Enter an incremental value.
e.g. X 1 0 0 Y 1 0 0 Z 0 END □	Enter the MAX point coordinates for the X, Y and Z axes; confirm the block with END.

The entered program section appears on the TNC screen:

```
% 743 G71 ❋
```

Block 1: Program beginning, name, unit of measure.

```
N10 G30 G17 X+0 Y+0 Z-40 ❋
```

Block 2: Spindle axis, MIN point coordinates.

```
N20 G31 G90 X+100 Y+100 Z+0 ❋
```

Block 3: MAX point coordinates.

```
N9999 % 743 G71 ❋
```

Block 4: Program end, name, unit of measure.

The unit of measure used in the program appears behind the program name (G71 = mm).

4.5 Entering Tool-Related Data

Besides the tool data and compensation, you must also enter the following information:

- Feed rate F
- Spindle speed S
- Miscellaneous functions M

The tool-related data can be determined with the aid of diagrams (see page 11-15).

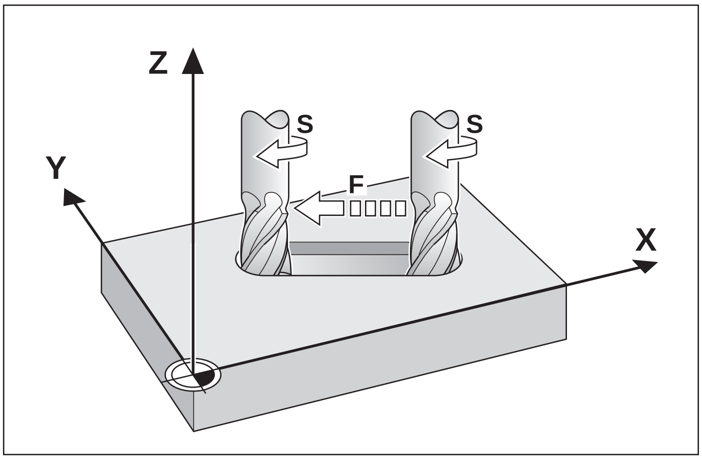


Fig. 4.11: Feed rate F and spindle speed S of the tool

Feed rate F

The feed rate is the speed in mm/min (or inch/min) with which the tool center moves.

Input range:
F = 0 to 30 000 mm/min (1181 inch/min)

The maximum feed rate is set in machine parameters individually for each axis.

To set the feed rate:

F ▶	e.g. 1 0 0	Enter the feed rate F, for example F = 100 mm/min.

Rapid traverse

You can program rapid traverse directly with the G00 function.

Duration of feed rate F

A feed rate that is entered as a numerical value remains in effect until the control executes a block in which another feed rate has been programmed.

If the new feed rate is G00 (rapid traverse), the feed rate will return to the last numerically entered feed rate as soon as the next block with G01 is executed.

Changing the feed rate F

You can vary the feed rate by turning the knob for feed rate override on the operating panel (see page 2-6).

Spindle speed S

The spindle speed S is entered in revolutions per minute (rpm).

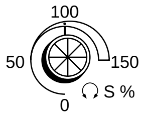
Input range:
S = 0 to 99 999 rpm

To change the spindle speed S in the part program:

S ▶	e.g. 1 0 0 0 END □	Enter the spindle speed S, for example 1000 rpm.

Resulting NC block: T1 G17 S1000

To change the spindle speed S during program run:

	You can vary the spindle speed S on machines with stepless ballscrew drives by turning the override knob on the operating panel.

4.6 Entering Miscellaneous Functions and STOP

The M functions (M for miscellaneous) affect:

- Program run
- Machine functions
- Tool behavior

On the inside back cover of this manual you will find a list of M functions that are predetermined for the TNC. The list indicates whether an M function begins at the start or at the end of the block in which it is programmed.

You can program several M functions in one NC block as long as they are independent of each other. The M function list on the inside back cover of this manual shows the different groups of M functions.

Some M functions are not effective on certain machines. The machine tool builder may also add some of his own M functions.

A program run or test run is interrupted when it reaches an NC block containing the function G38.

If you wish to interrupt the program run or test run for a certain duration, use the cycle G04: DWELL TIME (see page 8-38).

4.7 Actual Position Capture

Sometimes you may want to enter the actual position of the tool in a specific axis as a coordinate in a part program. Instead of reading the actual position values and entering them with the numeric keypad, you can simply press the "actual position capture" key. This feature can be used, for example, to enter the tool length.

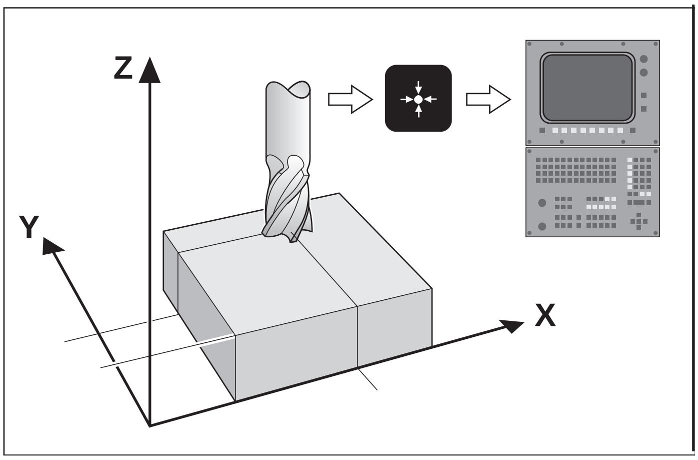


Fig. 4.12: Storing the actual position in the TNC

To capture the actual position:

▶ **MANUAL OPERATION**

Move the tool to the position that you wish to capture.

▶ **PROGRAMMING AND EDITING**

Select or create the program block in which you wish to enter the actual position of the tool.

e.g. X	Select the axis in which you wish to capture a coordinate, for example the X axis.
⌖	Transfer the actual position coordinate to the program.

Enter the radius compensation according to the position of the tool relative to the workpiece.

5.1 General Information on Programming Tool Movements

A tool movement is always programmed as if the tool is moving and the workpiece is stationary.

Always pre-position the tool at the beginning of a part program to prevent the possibility of damaging the tool or workpiece. In addition, radius compensation and a path function must be active.

*Example of an NC block: N30 G00 G40 G90 Z+100 **

Path functions

Each element of the workpiece contour is entered separately using path functions. The various path functions produce:

- Straight lines
- Circular arcs

You can also program a combination of the two (helical paths).

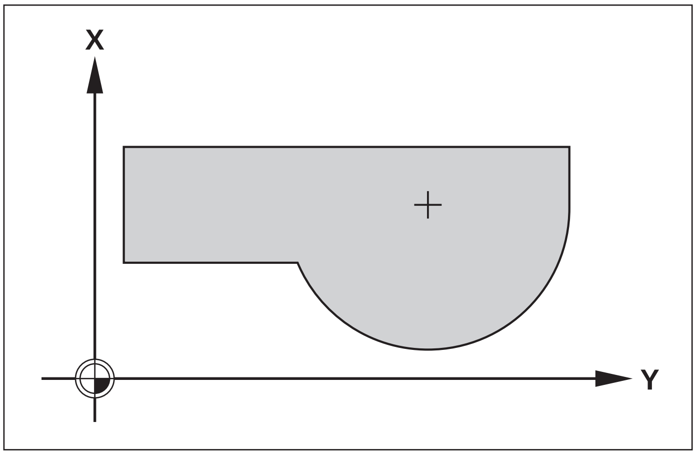


Fig. 5.1: A contour consists of a combination of straight lines and circular arcs

The contour elements are executed in sequence to machine the programmed contour.

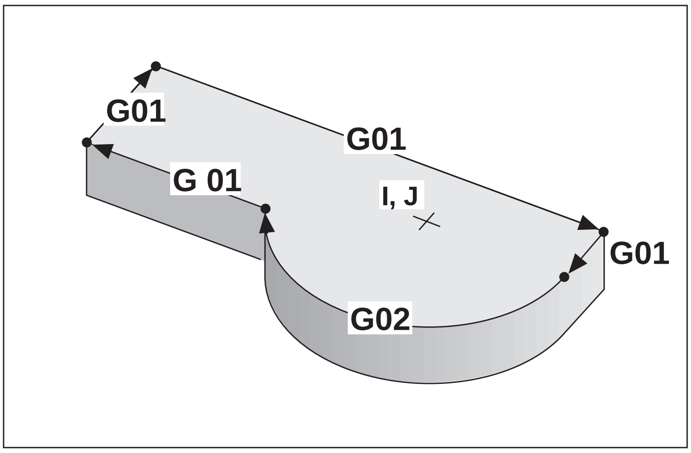


Fig. 5.2: Contour elements are programmed and executed in sequence

Subprograms and program section repeats

If a machining sequence repeats itself in a program, you can enter the sequence once and define it as a subprogram or program section repeat.

Programming possibilities:

- To repeat a machining routine immediately after it is executed (program section repeat)
- To insert a machining routine at certain locations in a program (subprogram)
- To call a separate program for execution or test run within the main program (main program as subprogram)

Cycles

Common machining routines are delivered with the control as standard cycles. The TNC features fixed cycles for:

- Pecking
- Tapping
- Slot milling
- Pocket and island milling

Coordinate transformation cycles can be used to change the coordinates of a machining sequence in a defined way, i.e.:

- Datum shift
- Mirroring
- Rotation
- Enlarging / Reducing

Parameter programming

Instead of numerical values you enter markers in the program, so-called parameters, which are defined through mathematical functions or logical comparisons. You can use parametric programming for:

- Conditional and unconditional jumps
- Measurements with the 3D touch probe during program run
- Output of values and messages
- Transferring values to and from memory

The following mathematical functions are available:

- Assign
- Addition/Subtraction
- Multiplication/Division
- Angle measurement/Trigonometry

etc.

5.2 Contour Approach and Departure

An especially convenient way to approach and depart a workpiece is on a tangential arc. This is done with the "smooth approach" function (G26) (see page 5-6).

Starting and end positions

Starting position

The tool moves from the starting position to the first contour point. The starting position is programmed without radius compensation.

The starting position must be:

- approachable without collision
- near the first contour point
- located to prevent contour damage during workpiece approach

If you choose a starting position within the hatch marked area of Fig. 5.3 the tool will damage the contour as it approaches the first contour point.

The best starting position Ⓢ lies on the extension of the tool path for machining the first contour element.

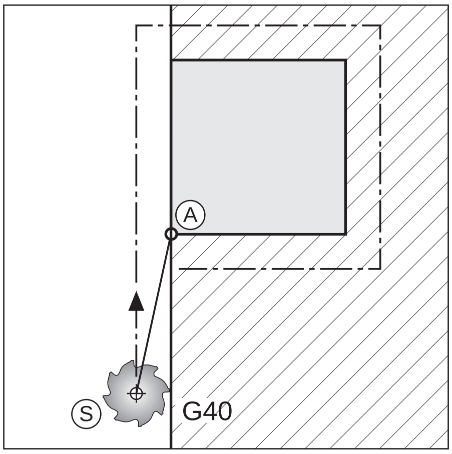


Fig. 5.3 : Starting position Ⓢ for contour approach

First contour point

Workpiece machining starts at the first contour point. The tool moves on a radius-compensated path to this point.

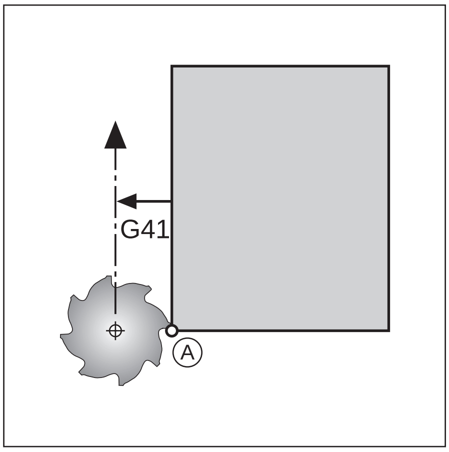


Fig. 5.4 : First contour point Ⓐ for machining

Approaching the starting point in the spindle axis

The spindle moves to its working depth as it approaches the starting position Ⓢ.

If there is any danger of collision, move the spindle axis separately to the starting position.

Example:	G00 G40	X ... Y ...	Positioning in X/Y
		Z–10	Positioning in Z

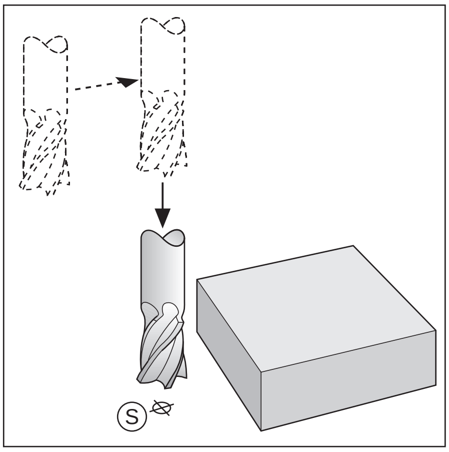


Fig. 5.5 : Move the spindle axis separately if there is any danger of collision

End position

The end position, like the starting position, must be

- approachable without collision
- near the last contour point
- located to prevent contour damage during workpiece departure

The best end position Ⓔ lies on the extension of the tool path. The end position can be located anywhere outside of the hatch marked area in Fig. 5.6. It is approached without radius compensation.

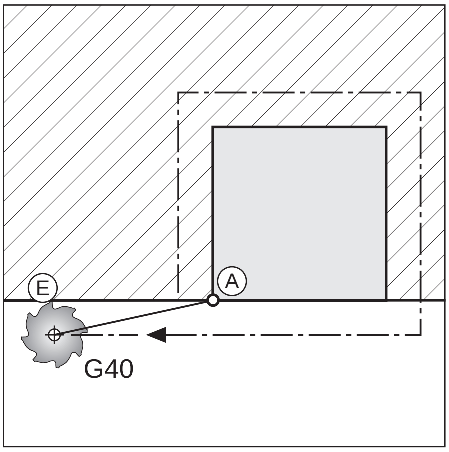


Fig. 5.6 : End position Ⓔ after machining

Departing the end position in the spindle axis

The spindle axis is moved separately when the end position is departed.

Example:	G00 G40	X ... Y ...	approaching the end position
		Z+50	retracting the tool

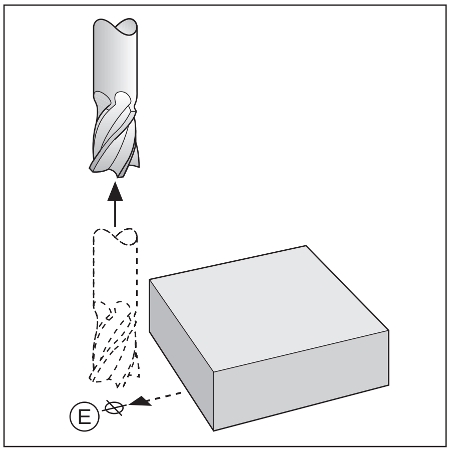


Fig. 5.7 : Retract separately in the spindle axis

Common starting and end position

A common starting and end position Ⓢⓔ can be located outside of the hatch marked area in the figures.

The best common starting and end position lies exactly between the extensions of the tool paths for machining the first and last contour elements.

A common starting and end position is approached without radius compensation.

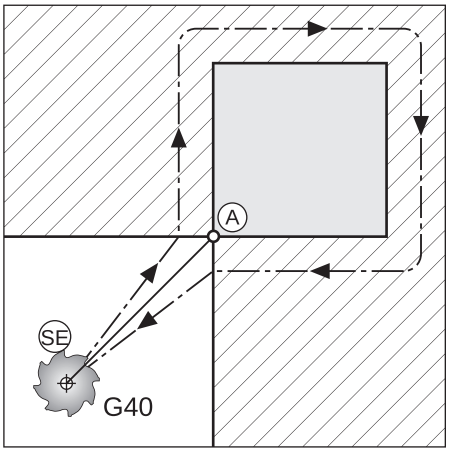


Fig. 5.8: Common starting and end position

Smooth approach and departure

The tool approaches and departs the workpiece at a tangent if you select the function G26 for approach and the function G27 for departure. This prevents dwell marks on the workpiece surface.

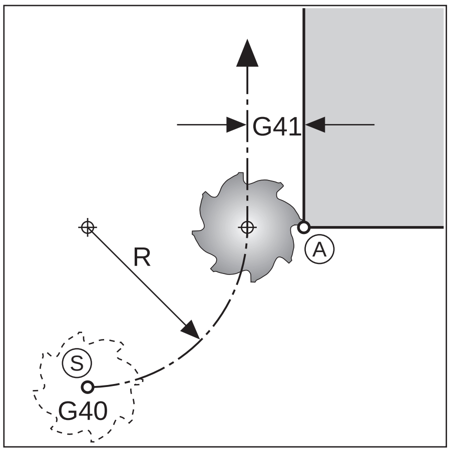


Fig. 5.9: Smooth approach onto a contour

Starting and end positions

The starting Ⓢ and end Ⓔ positions of machining lie outside of the workpiece and near the first and last contour elements, respectively.

The tool paths to the starting and end positions are programmed without radius compensation.

Input

- During contour approach, the function G26 is entered after the block in which the first contour point is programmed, i.e. after the first block with radius compensation G41/G42.
- During contour departure, the function G27 is entered after the block in which the last contour point is programmed, i.e. after the last block with radius compensation G41/G42.

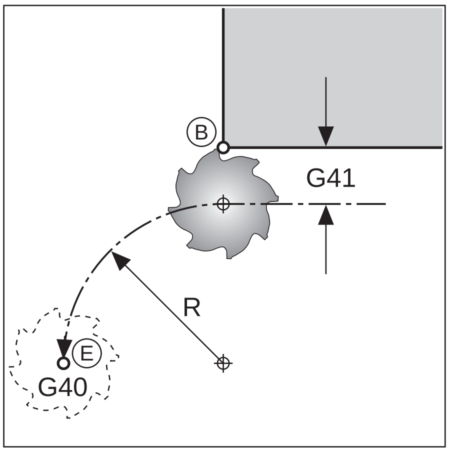


Fig. 5.10: Smooth departure from a contour

Program example

```
.
.
.
G00 G40 G90 X ... Y ... ............................................... Starting position Ⓢ
G01 G41 X ... Y ... F350 ............................................. First contour point Ⓐ
G26 R ................................................................... Smooth approach
.
.
.
Contour elements
.
.
.
X ... Y ...................................................................... Last contour point Ⓑ
G27 R ................................................................... Smooth departure
G00 G40 X ... Y ...................................................... End position Ⓔ
```

For proper execution of the functions G26/G27, a radius must be chosen such that the arc can connect the starting or end position with the contour point.

5.3 Path Functions

General information

Part program input

To create a part program you enter the dimensional information given on the workpiece drawing. The workpiece coordinates are programmed as absolute values (G90) or as relative values (G91).

You usually program a contour element by entering its end point. The TNC automatically calculates the tool path from the tool data and the radius compensation.

The first machining block after the tool call must contain the following G functions:

Path function	e.g. G00
Radius compensation	e.g. G40
Absolute or incremental programming	e.g. G90

Machine axis movement under program control

All machine axes programmed in a single NC block are moved simultaneously.

Paraxial movement

Paraxial movement means that the tool path is parallel to the programmed axis.

Number of axes programmed in the NC block: 1

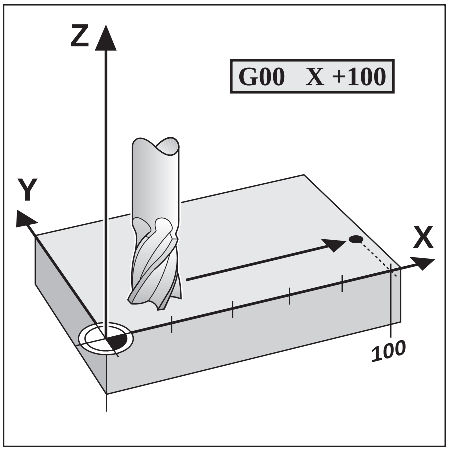


Fig. 5.11: Paraxial movement

Movement in the main planes

With this type of movement the tool moves to the programmed position on a straight line or a circular arc in a "working plane".

Number of axes programmed in the NC block: 2

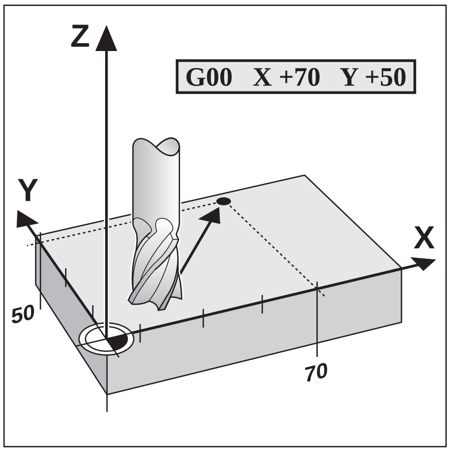


Fig. 5.12: Movement in a main plane (X/Y plane)

Movement of three machine axes (3D movement)

The tool moves in a straight line to the programmed position.

Number of axes programmed in the NC block: 3

Exception: A helical path is created by combining a circular movement in a plane with a linear movement perpendicular to the plane.

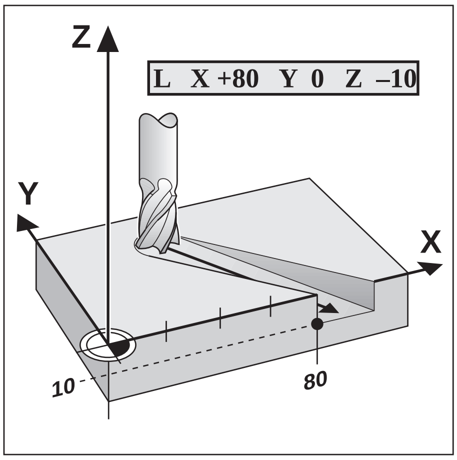


Fig. 5.13: Three-dimensional tool movement

Overview of path functions

Function	Input in Cartesian coordinates	 in polar coordinates
Straight line at rapid traverse.	G00	G10
Straight line with a programmed feed rate.	G01	G11
Chamfer with chamfer length R. A chamfer is inserted between two intersecting straight lines.	G24	
Circle center – at the same time a reference for polar coordinates. I,J,K do not generate a movement.	I, J, K	
Circular movement in the clockwise direction (CW).	G02	G12
Circular movement in the counterclockwise direction (CCW). A circular path can be programmed by entering: • Circle center I, J, K and end point, or • Circle radius and end point.	G03	G13
Circular path with no direction of rotation defined. The circular path is programmed by entering circle center and end point. The direction of rotation is taken from the last programmed circular movement (G02/G12 or G03/G13).	G05	G15
Circular movement with tangential connection. A circular arc is connected tangentially with the previously programmed contour element. The end point of the circular arc is entered in the part program.	G06	G16
Corner rounding with radius R. A circular arc is inserted to connect tangentially both with the preceding and the subsequent contour elements.	G25	

5.4 Path Contours - Cartesian Coordinates

Straight line at rapid traverse G00

Straight line with feed rate G01 F ...

To program a straight line, you enter:

- The coordinates of the end point Ⓔ
- If necessary:
  Radius compensation, feed rate, miscellaneous function

The tool moves in a straight line from its starting position to the end point Ⓔ. The starting position Ⓢ was reached in the previous block.

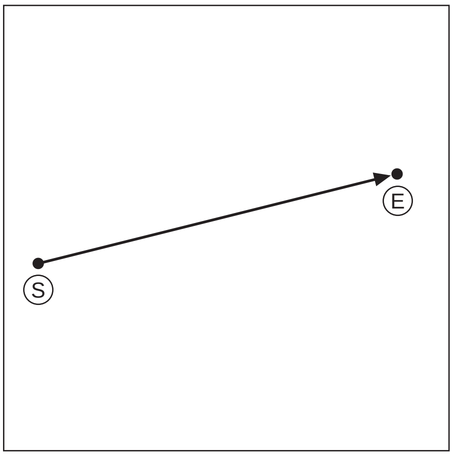


Fig. 5.14: A linear movement

To program a straight line:

G 0 0	Straight line at rapid traverse.
If necessary G 9 1	Identify coordinates as relative values, for example G91 X–50 mm.
e.g. X	Press the orange axis selection key, for example X.
e.g. 5 0	Enter the coordinate of the end point.
If necessary +/–	If the coordinate is negative, press the +/- key once, for example X = –50 mm.
e.g. Y ⋮ e.g. Z	Enter all further coordinates of the end point.

⋮

G 4 1	The tool must move to the left of the programmed contour to compensate its own radius.
G 4 2	The tool must move to the right of the programmed contour to compensate its own radius.
G 4 0	The tool moves directly to the end point.
M e.g. 3 ENT	Enter a miscellaneous function, for example M3 (spindle on, clockwise rotation).
END □	Conclude the block with END as soon as all coordinates are entered.

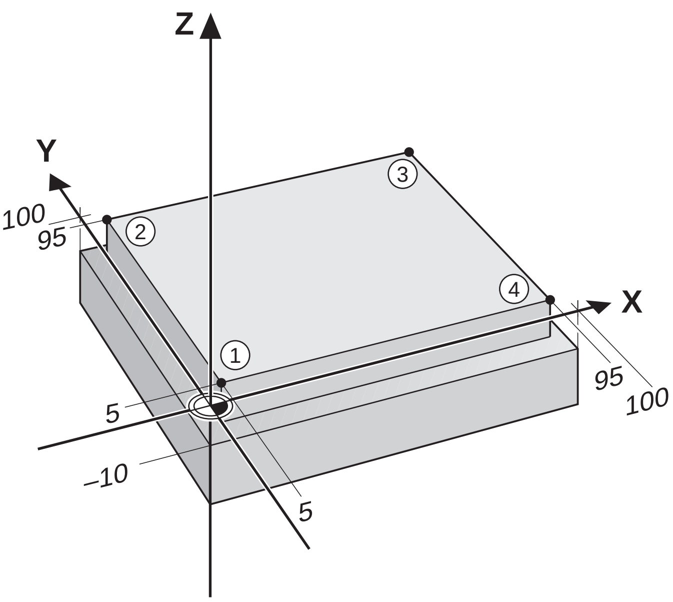
*Resulting NC block: N25 G00 G42 G91 X+50 G90 Y+10 Z–20 M3 **

Example for exercise: Milling a rectangle

Coordinates of the corner points:

①	X = 5 mm	Y = 5 mm
②	X = 5 mm	Y = 95 mm
③	X = 95 mm	Y = 95 mm
④	X = 95 mm	Y = 5 mm

Milling depth: Z = –10mm



Part program

```
%S512I G71 * .......................................................... Begin program; program name S512I;
                                                                         dimensions in millimeters
N10 G30 G17 X+0 Y+0 Z–20 *
N20 G31 G90 X+100 Y+100 Z+0 * ........................... Define blank form for graphic workpiece simulation
                                                                         (MIN and MAX point)
N30 G99 T1 L+0 R+5 * ............................................ Define tool in the program
N40 T1 G17 S2500 * ................................................ Call tool in the spindle axis Z (G17);
                                                                         spindle speed S = 2500 rpm
N50 G00 G40 G90 Z+100 M06 * .............................. Retract in the spindle axis; rapid traverse; miscellaneous
                                                                       function for tool change
N60 X–10 Y–10 *  ..................................................... Pre-position near the first contour point
N70 Z–10 M03 * ....................................................... Pre-position in Z; spindle on
N80 G01 G41 X+5 Y+5 F150 * ................................. Move to point ① with radius compensation
N90 Y+95 * ............................................................... Move to corner point ②
N100 X+95 * ............................................................. Move to corner point ③
N110 Y+5 * ............................................................... Move to corner point ④
N120 X+5 * ............................................................... Move to corner point ①, conclude milling
N130 G00 G40 X–10 Y–10 M05 * ............................. Retract in X and Y, cancel radius compensation,
                                                                      spindle STOP
N140 Z+100 M02 * .................................................... Move tool to setup clearance, spindle OFF, coolant OFF,
                                                                         program stop, return jump to block 1
N9999 %S512I G71 * ............................................... End of program
```

Chamfer G24

The chamfer function permits you to cut off corners at the intersection of two straight lines.

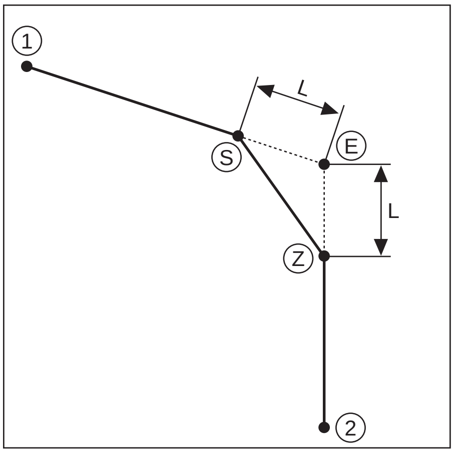


Fig. 5.15: Chamfer from Ⓢ to Ⓩ

You enter the length L to be removed from each side of the corner.

Prerequisites:

- The radius compensation before and after the chamfer block must be the same.
- An inside chamfer must be large enough to accommodate the current tool.

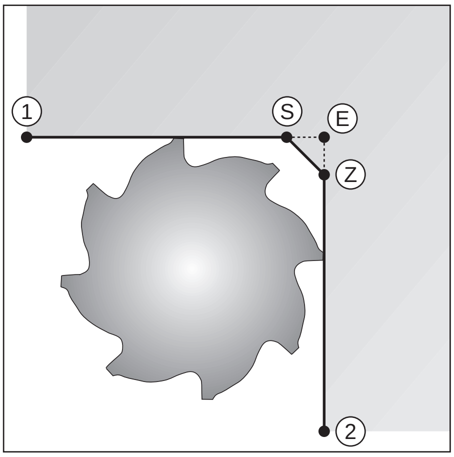


Fig. 5.16: Tool radius too large

- You cannot start a contour with a G24 block.
- A chamfer is only possible in the working plane.
- The feed rate for chamfering is taken from the previous block.
- The corner point E is cut off by the chamfer and is not part of the resulting contour.

To program a chamfer:

G 2 4 ENT	Select the chamfer function.

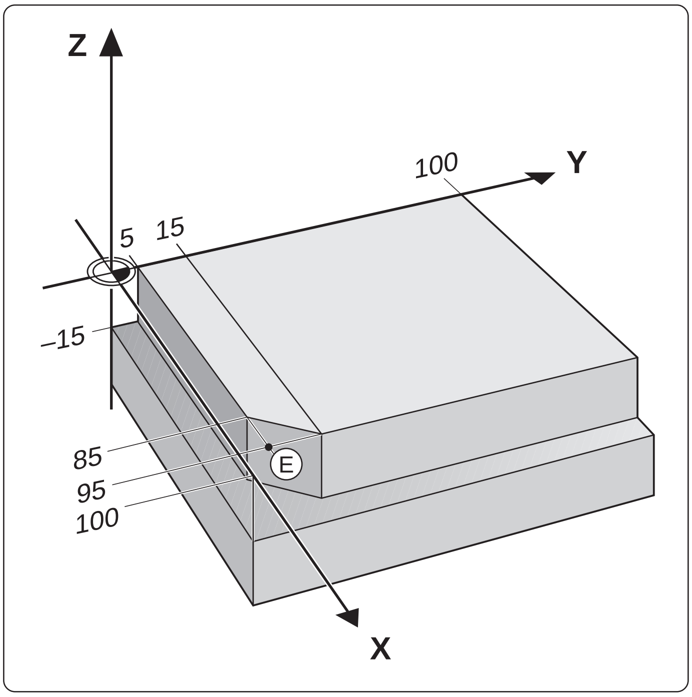
CHAMFER SIDE LENGTH?

e.g. 5 END □	Enter the length to be removed from each side of the corner, for example 5 mm.

*Resulting NC block: G24 R5 **

Example for exercise: Chamfering a corner

Coordinates of the corner point Ⓔ:	X	=	95 mm
	Y	=	5 mm
Chamfer length:	LF	=	10 mm
Milling depth:	Z	=	–15 mm
Tool radius:	R	=	+10 mm



Part program

```
%S514I G71 * ............................................ Begin program
N10 G30 G17 X+0 Y+0 Z–20 * .............................. Workpiece blank MIN point
N20 G31 G90 X+100 Y+100 Z+0 * ........................... Workpiece blank MAX point
N30 G99 T5 L+5 R+10 * ................................... Tool definition
N40 T5 G17 S2000 * ...................................... Tool call
N50 G00 G40 G90 Z+100 M06 * ............................. Retract spindle and insert tool
N60 X–10 Y–5 * .......................................... Pre-position in X, Y
N70 Z–15 M03 * .......................................... Pre-position to the working depth, spindle on
N80 G01 G42 X+5 Y+5 F200 * .............................. Move with radius compensation and reduced feed to
                                                          the first contour point
N90 X+95 * .............................................. Program the first straight line for corner E
N100 G24 R10 * .......................................... Chamfer block: inserts a chamfer with L = 10 mm
N110 Y+100 * ............................................ Program the second straight line for corner E
N120 G00 G40 X+110 Y+110 * .............................. Retract the tool in X, Y and Z, cancel radius
                                                          compensation
N130 Z+100 M02 * ........................................ Move tool to setup clearance
N9999 %S514I G71 *
```

Circles and circular arcs - General information

The TNC can control two machine axes simultaneously to move the tool in a circular path.

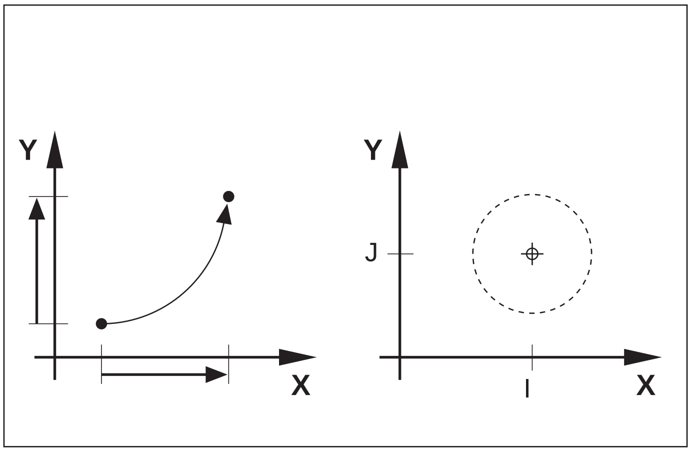


Fig. 5.17: Circular arc and circle center

Circle center I, J, K

You can define a circular movement by entering its center.

A circle center can also serve as reference (pole) for polar coordinates.

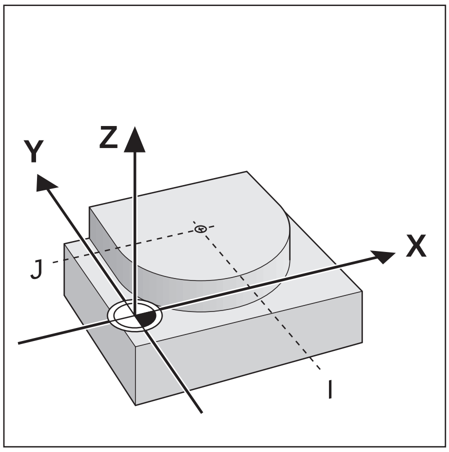


Fig. 5.18: Circle center coordinates

Direction of rotation

When there is no tangential transition to another contour element, enter the mathematical direction of rotation, where

- a clockwise direction of rotation is mathematically negative: function G02
- a counterclockwise direction of rotation is mathematically positive: function G03

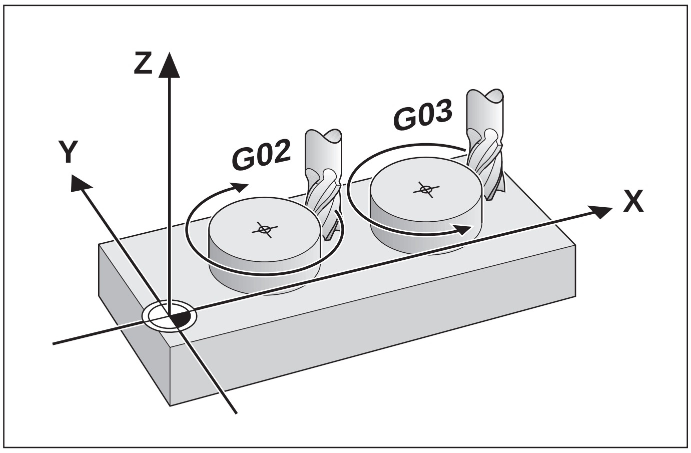


Fig. 5.19: Direction of rotation for circular movements

Radius compensation in circular paths

You cannot begin radius compensation in a circle block. It must be activated beforehand in a line block.

Circles in the main planes

When you program a circle, the TNC assigns it to one of the main planes. This plane is automatically defined when you set the spindle axis during tool call (T).

Spindle axis	Main plane	Circle center
Z	XY G17	I J
Y	ZX G18	K I
X	YZ G19	J K

Fig. 5.20: Defining the spindle axis also defines the main plane and the circle center designations

You can program circles that do not lie parallel to a main plane by using Q parameters (see Chapter 7).

Circle center I, J, K

If you program an arc using the functions G02/G03/G05, you must first define the circle center by:

- entering the Cartesian coordinates of the circle center
- using the circle center defined in an earlier block
- capturing the actual position

You can define the last programmed position as circle center/pole by programming G29.

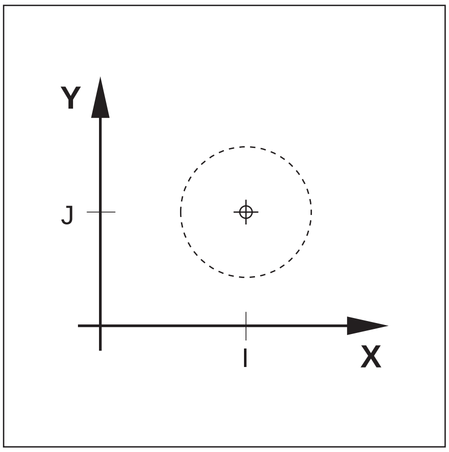


Fig. 5.21: Circle center I, J

Duration of a circle center definition

A circle center definition remains effective until a new circle center is defined.

Entering I, J, K in relative values

If you enter the circle center with relative coordinates, you have defined it relative to the last programmed tool position.

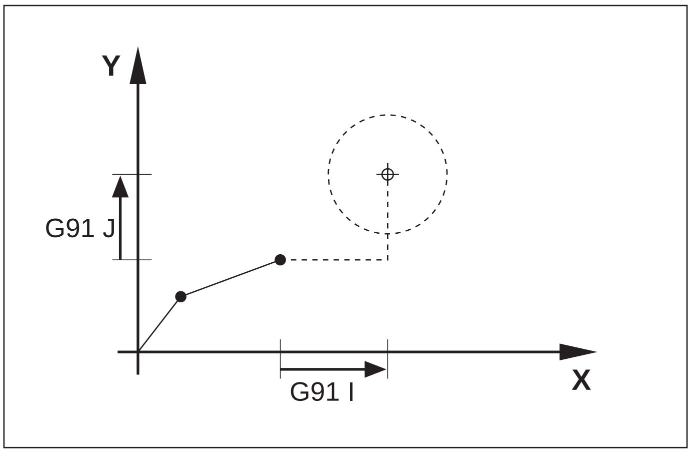


Fig. 5.22: Incremental circle center coordinates

- The circle center I, J, K also serves as pole for polar coordinates.
- I, J, K defines a position as a circle center. The resulting contour is located on the circle, not on the circle center.

To program a circle center (pole):

e.g. I e.g. 2 0	Select the coordinate axis for the circle center. Enter the coordinate for the circle center on this axis, for example I = 20 mm.
e.g. J e.g. 1 0 +/– END □	Select the second coordinate axis, for example J. Enter the coordinate of the circle center, for example J = –10 mm.

*Resulting NC block: I+20 J–10 **

Circular path G02/G03/G05 around the circle center I, J, K

Prerequisites

The circle center I, J, K must have been previously defined in the program. The tool is located at the arc starting point Ⓢ.

Defining the direction of rotation

You can program the following directions of rotation:

• Clockwise rotation	G02
• Counterclockwise rotation	G03
• No direction of rotation defined The tool moves in the direction of rotation defined in an earlier block.	G05

Input

- Arc end point

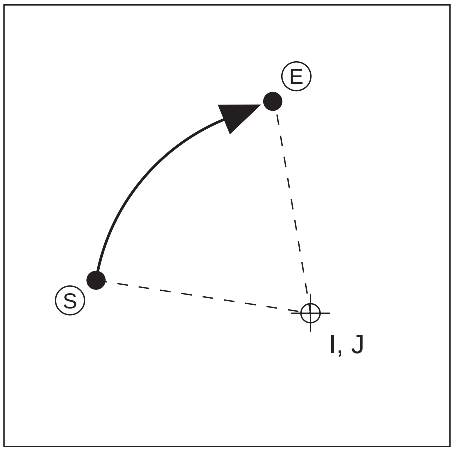


Fig. 5.23: A circular arc from Ⓢ to Ⓔ around I, J

> The starting and end points of the arc must lie on the circle.
> Input tolerance: up to 0.016 mm

- To program a full circle, enter the same position for the end point as for the starting point in a G02/G03 block.

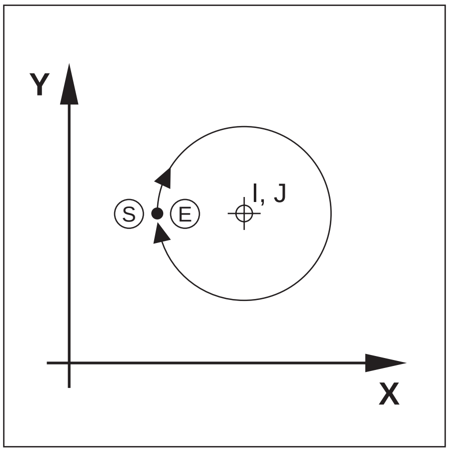


Fig. 5.24: Full circle around I, J with a G02 block

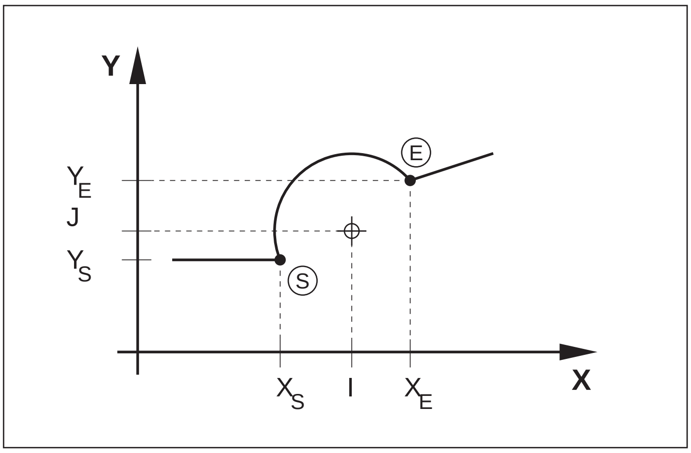


Fig. 5.25: Coordinates of a circular arc

To program a circular arc around a circle center I, J with G02 (direction of rotation = clockwise):

G 0 2	Program the circle with Cartesian coordinates and clockwise rotation.
G 9 1 X 5	Enter the first coordinate of the end point as an incremental value, for example X = 5 mm.
G 9 0 Y 5 +/–	Enter the second coordinate of the end point as an absolute value, for example Y = –5 mm.
END □	Terminate the block.

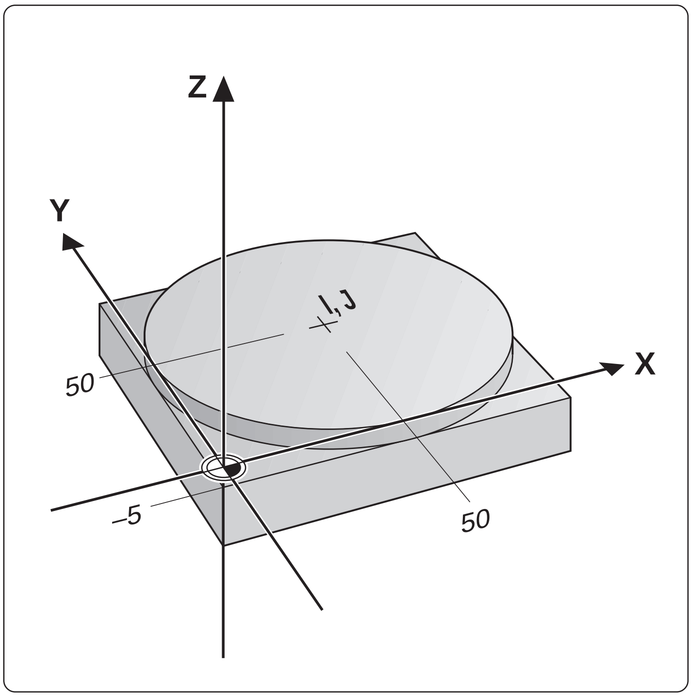
If necessary, enter also:

- Radius compensation
- Feed rate
- Miscellaneous function

Resulting NC block: G02 G91 X+5 G90 Y–5

Example for exercise: Milling a full circle in one block

Circle center:	I	=	50 mm
	J	=	50 mm
Beginning and end of the circular arc:	X	=	50 mm
	Y	=	0 mm
Milling depth:	Z	=	– 5 mm
Tool radius:	R	=	15 mm



Part program

```
%S520I G71 * ........................................................... Begin program
N10 G30 G17 X+1 Y+1 Z–20 * ................................. Workpiece blank MIN point
N20 G31 G90 X+100 Y+100 Z+0 * ........................... Workpiece blank MAX point
N30 G99 T6 L+0 R+15 * ........................................... Tool definition
N40 T6 G17 S1500 * ................................................. Tool call
N50 G00 G40 G90 Z+100 M06 * .............................. Retract spindle and insert tool
N60 X+50 Y–40 * ...................................................... Pre-position in X, Y
N70 Z-5 M03 * .......................................................... Pre-position to the working depth
N80 I+50 J+50 * ....................................................... Coordinates of the circle center
N90 G01 G41 X+50 Y+0 F100 * ............................... Move with radius compensation and reduced feed to the
                                                             first contour point
N100 G26 R10 * ....................................................... Smooth (tangential) approach
N110 G02 X+50 Y+0 * .............................................. Mill circular arc around circle center I,J; negative direction
                                                             of rotation; end point coordinates X = +50 und Y = +0
N120 G27 R10 * ....................................................... Smooth (tangential) departure
N130 G00 G40 X+50 Y–40 * ..................................... Retract the tool in X, Y; cancel radius compensation
N140 Z+100 M02 * .................................................... Retract the tool in Z
N9999 %S520I G71 *
```

Circular path G02/G03/G05 with defined radius

The tool moves on a circular path with the radius R.

Defining the direction of rotation

• Clockwise rotation	G02
• Counterclockwise rotation	G03
• No direction of rotation defined The tool moves in the direction of rotation defined in an earlier block.	G05

Input

- Coordinates of the arc end point
- Arc radius R

- To program a full circle you must enter two successive G02/G03 blocks.
- The distance from the starting point to the end point cannot be larger than the diameter of the circle.
- The maximum permissible radius is 30 m (9.8 ft).

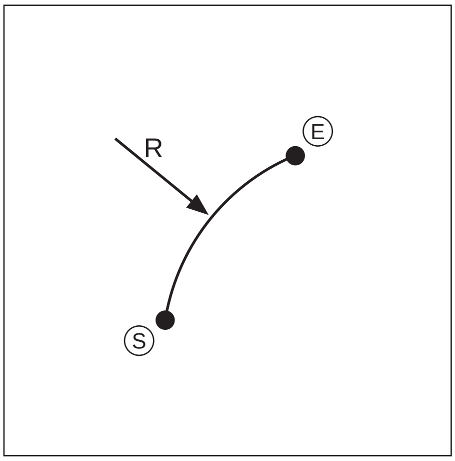


Fig. 5.26: Circular path from Ⓢ to Ⓔ with radius R

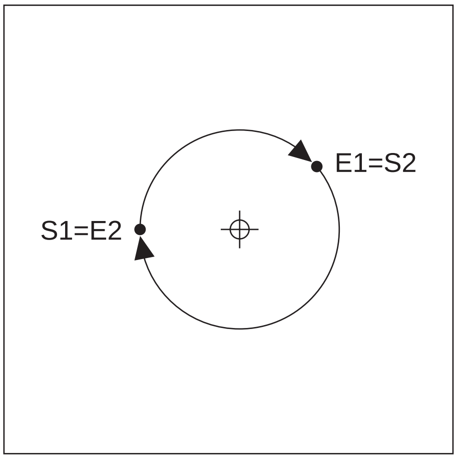


Fig. 5.27: Full circle with two G02 blocks

Central angle CCA and arc radius R

Starting point Ⓢ and end point Ⓔ can be connected by four different arcs with the same radius. The arcs differ in their curvatures and lengths.

Large circular arc: CCA>180° (the circular arc is longer than a semicircle)
Input: radius R with negative sign (R<0).

Small circular arc: CCA<180° (the circular arc is shorter than a semicircle)
Input: radius R with positive sign (R>0).

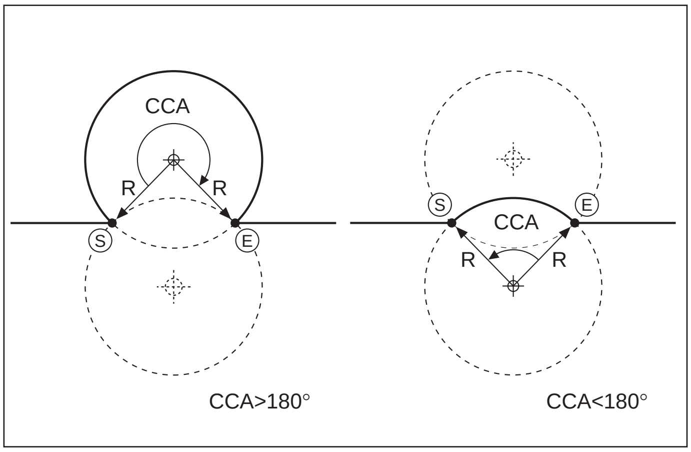


Fig. 5.28: Circular arcs with central angles greater than and less than 180°

Direction of rotation and arc shape

This direction of rotation determines whether the arc is

- convex (curved outward) or

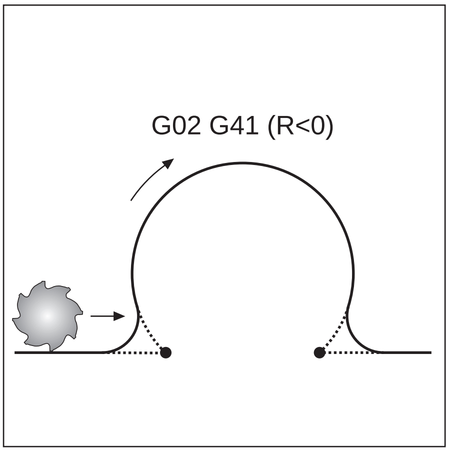


Fig. 5.29: Convex path

- concave (curved inward)

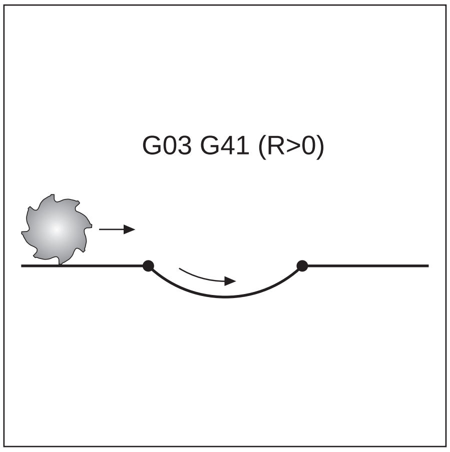


Fig. 5.30: Concave path

To program a circular arc with defined radius:

G 0 2	Program the circle with Cartesian coordinates and clockwise rotation.
e.g. X 1 0 Y 2	Enter the coordinates of the arc end point, for example X = 10 mm, Y = 2 mm.
e.g. R 5 +/– END □	Enter the arc radius, for example R = 5 mm, and determine the size of the arc using the algebraic sign, here the negative sign.

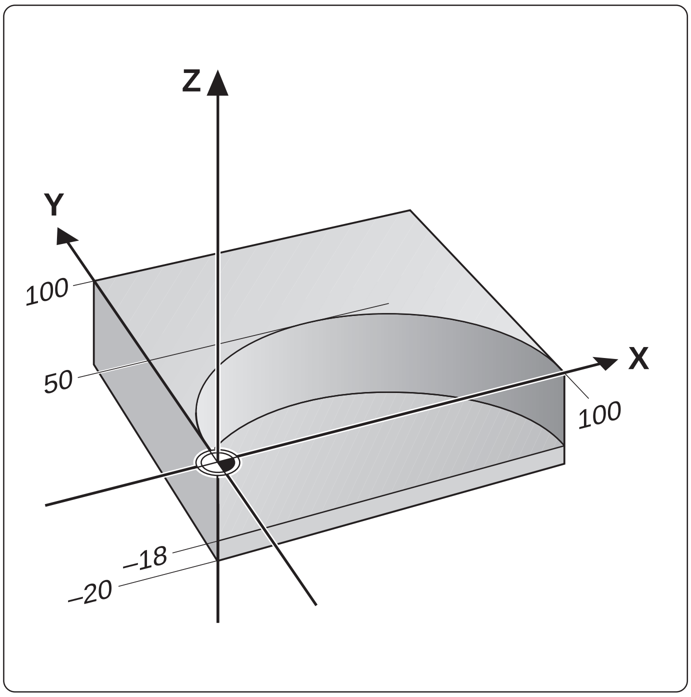
If necessary, enter also:

- Radius compensation
- Feed rate
- Miscellaneous function

Resulting NC block: G02 G41 X+10 Y+2 R–5

Example for exercise: Milling a concave semicircle

Semicircle radius:	R	=	50 mm
Coordinates of the arc starting point:	X	=	0
	Y	=	0
Coordinates of the arc end point:	X	=	100 mm
	Y	=	0
Tool radius:	R	=	25 mm
Milling depth:	Z	=	–18 mm



Part program

```
%S523I G71 * ............................................................ Begin program
N10 G30 G17 X+0 Y+0 Z–20 * ................................. Define the workpiece blank
N20 G31 G90 X+100 Y+100 Z+0 *
N30 G99 T1 L+0 R+25 * ............................................ Define the tool
N40 T1 G17 S780 * .................................................... Call the tool
N50 G00 G40 G90 Z+100 M06 * ............................... Retract the spindle and insert the tool
N60 X+25 Y-30 * ........................................................ Pre-position in X, Y
N70 Z–18 M03 * ........................................................ Pre-position to the working depth
N80 G01 G42 X+0 Y+0 F100 * ................................. Move with radius compensation and reduced feed to
                                                                  the first contour point
N90 G02 X+100 Y+0 R–50 * ...................................... Mill circular arc to the end point X = 100, Y = 0;
                                                                  radius = 50, negative direction of rotation
N100 G00 G40 X+70 Y–30 * ...................................... Retract the tool in X, Y; cancel radius compensation
N110 Z+100 M02 * .................................................... Retract the tool in Z
N9999 %S523I G71 *
```

Circular path G06 with tangential connection

The tool moves in an arc that starts at a tangent with the previously programmed contour.

A transition between two contour elements is called tangential when one contour element makes a smooth and continuous transition to the next. There is no visible corner at the intersection.

Input

Coordinates of the arc end point.

Prerequisites

- The contour element to which the tangential arc connects with G06 must be programmed immediately before the G06 block.
- There must be at least two positioning blocks defining the tangentially connected contour element before the G06 block.

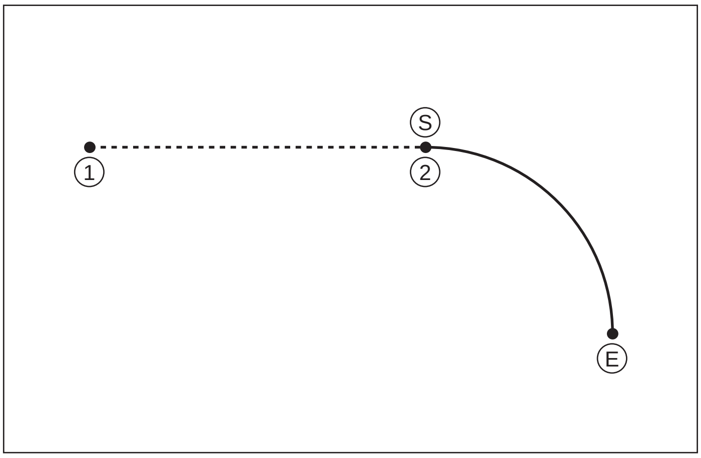


Fig. 5.31: The straight line ① - ② is connected tangentially to the circular arc Ⓢ - Ⓔ.

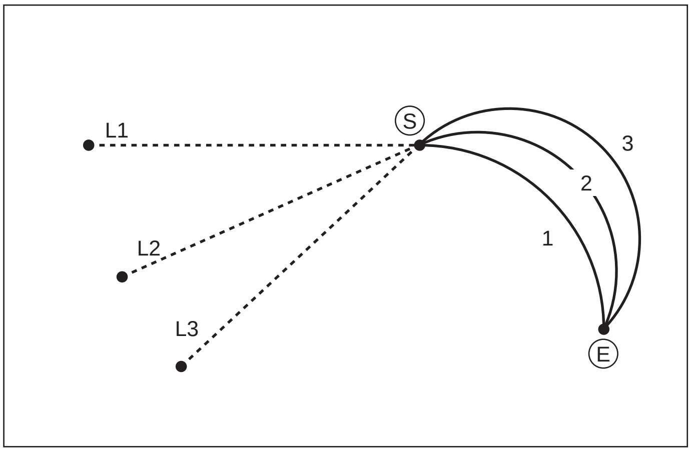


Fig. 5.32: The path of a tangential arc depends on the preceding contour element.

A tangential arc is a two-dimensional operation: the coordinates in the G06 block and the positioning block before it should be in the plane of the arc.

To program a circular path G06 with tangential connection:

G 0 6	Circular path with tangential connection.
G 9 1 X 5 0 Y +/– 1 0 END □	Enter the coordinates of the arc end point as relative coordinates, for example X = 50 mm, Y = –10 mm.

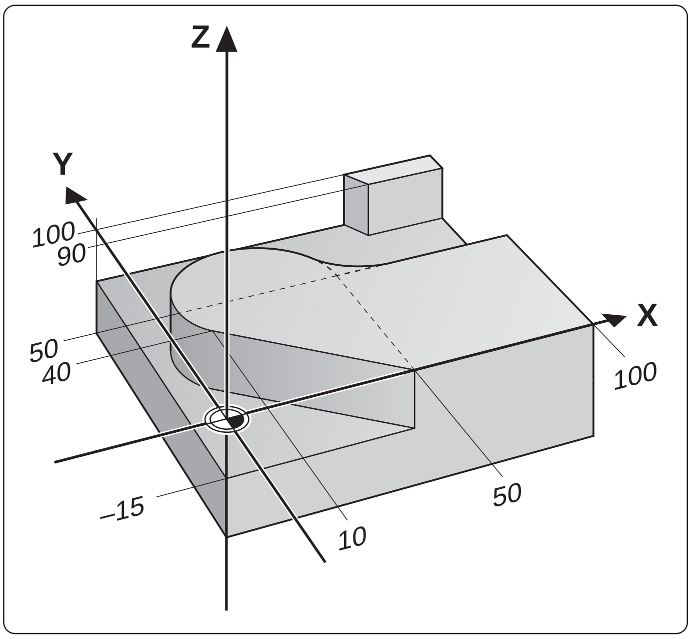
If necessary, enter also:

- Radius compensation
- Feed rate
- Miscellaneous function

*Resulting NC block: G06 G42 G91 X+50 Y–10 **

Example for exercise: Circular arc connecting to a straight line

Coordinates of the transition point from the line to the arc:	X	=	10 mm
	Y	=	40 mm
Coordinates of the arc end point:	X	=	50 mm
	Y	=	50 mm
Milling depth:	Z	=	–15 mm
Tool radius:	R	=	20 mm



Part program

```
%S525I G71 * ........................................................... Begin program
N10 G30 G17 X+0 Y+0 Z–20 * ................................. Define workpiece blank
N20 G31 G90 X+100 Y+100 Z+0 *
N30 G99 T12 L-25 R+20 * ........................................ Define the tool
N40 T12 G17 S1000 * ................................................ Call the tool
N50 G00 G40 G90 Z+100 M06 * .............................. Retract the spindle and insert the tool
N60 X+30 Y–30 * ...................................................... Pre-position in X, Y
N70 Z–15 M03 * ........................................................ Pre-position to the working depth
N80 G01 G41 X+50 Y+0 F100 * ............................... Move with radius compensation and reduced
                                                               feed to the first contour point
N90 X+10 Y+40 * ..................................................... Straight line connecting tangentially to the arc
N100 G06 X+50 Y+50 *                                           Arc to end point with coordinates X = 50 and Y = 50;
                                                               connects tangentially to the straight line in block N90
N110 G01 X+100 * ................................................... End of contour
N120 G00 G40 X+130 Y+70 * .................................. Retract the tool in X, Y; cancel radius compensation
N130 Z+100 M02 * ................................................... Retract the tool in Z
N9999 %S525I G71 *
```

Corner rounding G25

The tool moves in an arc that connects tangentially both with the preceding and the subsequent contour elements.

The function G25 is useful for rounding corners.

Input

- Radius of the arc
- Feed rate for the arc

Prerequisite

On inside corners, the rounding arc must be large enough to accommodate the tool.

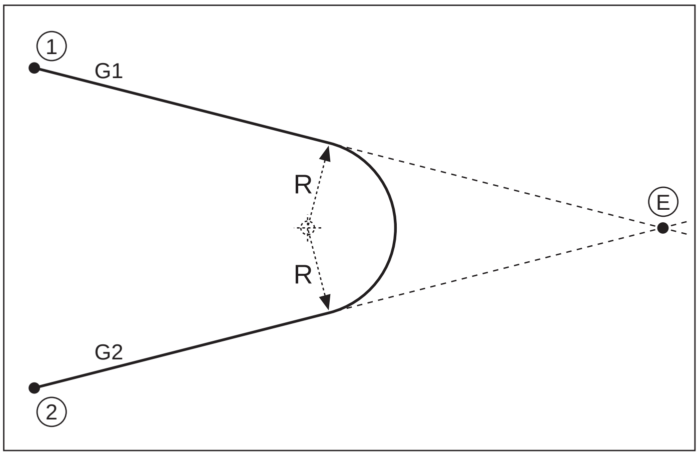


Fig. 5.33: Rounding radius R between G1 and G2

- In the preceding and subsequent positioning blocks both coordinates should lie in the plane of the arc.
- The corner point Ⓔ is cut off by the rounding arc and is not part of the contour.
- A feed rate programmed in the G25 block is effective only in that block. After the G25 block the previous feed rate becomes effective again.

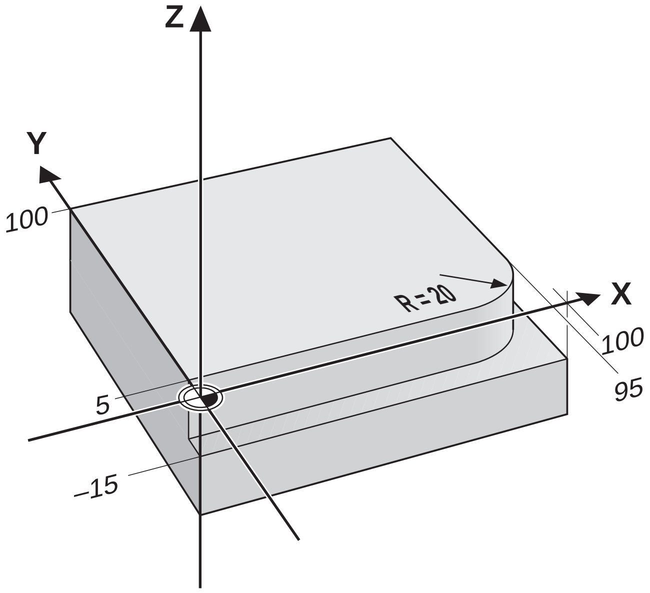
To program a tangential arc between two contour elements:

G 2 5 ENT	Select corner rounding.
ROUNDING RADIUS	
e.g. 1 0 ENT	Enter the rounding radius, for example R = 10 mm.
e.g. 1 0 0 END □	Enter the feed rate for the rounding radius, for example F = 100 mm/min.

Resulting NC block: G25 R 10 F 100

Example for exercise: Rounding a corner

Coordinates of the corner point:	X	=	95	mm
	Y	=	5	mm
Rounding radius:	R	=	20	mm
Milling depth:	Z	=	–15	mm
Tool radius:	R	=	10	mm



Part program

```
%S527I G71 * ............................................................ Begin program
N10 G30 G17 X+0 Y+0 Z–20 * ................................. Define the workpiece blank
N20 G31 G90 X+100 Y+100 Z+0 *
N30 G99 T7 L+0 R+10 * ............................................ Define the tool
N40 T7 G17 S1500 * ................................................. Call the tool
N50 G00 G40 G90 Z+100 M06 * ............................... Retract the spindle and insert the tool
N60 X–10 Y-5 * .......................................................... Pre-position in X, Y
N70 Z–15 M03 * ......................................................... Pre-position to the working depth
N80 G01 G42 X+0 Y+5 F100 *                                Move with radius compensation and reduced feed to
the first contour element
N90 X+95 * ................................................................. Program the first straight line for the corner
N100 G25 R20 * ......................................................... Insert radius R = 20 mm between the two contour elements
N110 Y+100 * ............................................................ Program the second straight line for the corner
N120 G00 G40 X+120 Y+120 * ................................. Retract the tool in X, Y; cancel radius compensation
N130 Z+100 M02 * .................................................... Retract the tool in Z
N9999 %S527I G71 *
```

5.5 Path Contours - Polar Coordinates

Polar coordinates are useful for programming:

- Positions on circular arcs
- Positions from workpiece drawings showing angular dimensions

Section 1.2 "Fundamentals of NC" (see page 1-8) provides a detailed description of polar coordinates.

Polar coordinate origin: Pole I, J, K

You can define the pole anywhere in the program before the blocks containing polar coordinates. Enter the pole in Cartesian coordinates as a circle center in a I, J, K block. A pole definition remains effective until a new pole is defined. The designation of the pole is derived from its position in the working plane.

Working plane	POLE
XY	I, J
YZ	J, K
ZX	K, I

You can define the last programmed position as POLE by entering G29.

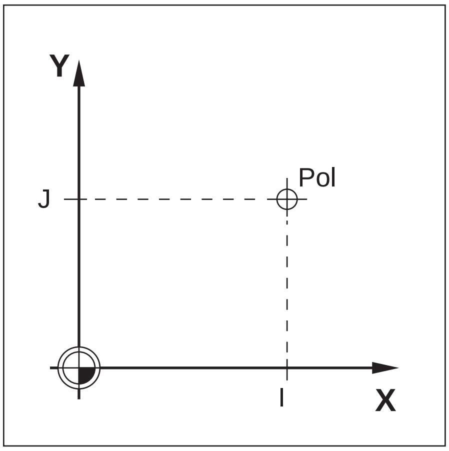


Fig. 5.34: The pole is entered as circle center

Straight line at rapid traverse G10

Straight line with feed rate G11 F ...

- You can enter any value from –360° to +360° for H.
- Enter the algebraic sign for H relative to the angle reference axis:
  For an angle from the reference axis counterclockwise to R: H>0
  For an angle from the reference axis clockwise to R: H<0

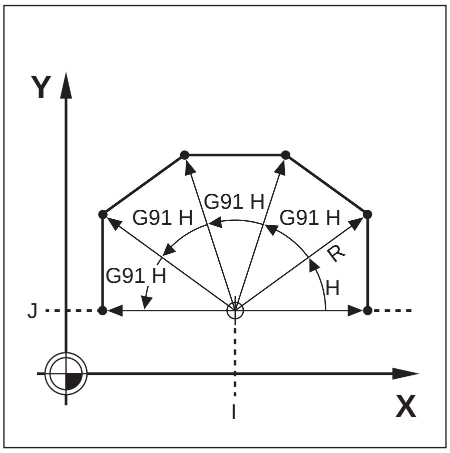


Fig. 5.35: Contour consisting of straight lines with polar coordinates

G 1 0	Straight line with polar coordinates at rapid traverse.
R 5	Enter the radius from the pole to the straight line end point, for example PR = 5 mm.
H 3 0 END □	Enter the angle from the angle reference axis to R, for example H = 30°.

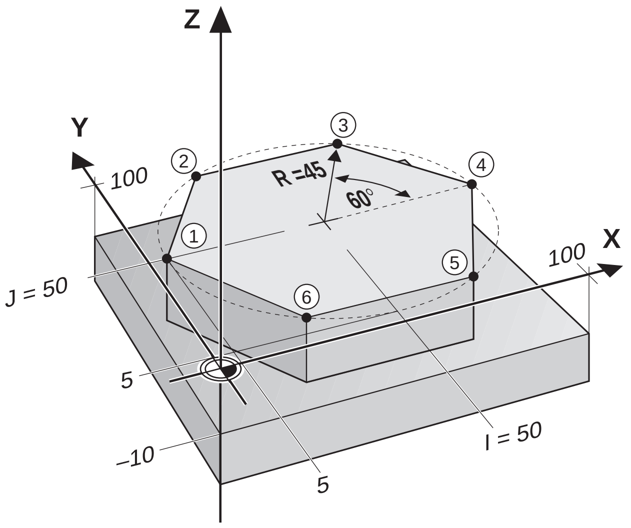
*Resulting NC block: G10 R5 H30 **

Example for exercise: Milling a hexagon

Corner point coordinates:

①	H	=	180°	R	=	45 mm
②	H	=	120°	R	=	45 mm
③	H	=	60°	R	=	45 mm
④	H	=	0°	R	=	45 mm
⑤	H	=	300°	R	=	45 mm
⑥	H	=	240°	R	=	45 mm

Milling depth:	Z	=	–10 mm
Tool radius:	R	=	5 mm



Part program

```
%S530I G71 * ............................................................ Begin program
N10 G30 G17 X+0 Y+0 Z–20 * .................................. Define the workpiece blank
N20 G31 G90 X+100 Y+100 Z+0 *
N30 G99 T1 L+0 R+17 * ............................................ Define the tool
N40 T1 G17 S3200 * .................................................. Call the tool
N50 G00 G40 G90 Z+100 M06 * ............................... Retract the spindle and insert the tool
N60 I+50 J+50 * ......................................................... Set the pole
N70 G10 R+70 H–190 * ............................................. Pre-position in X, Y to polar coordinates
N80 Z–10 M03 * ........................................................ Pre-position to working depth
N90 G11 G41 R+45 H+180 F100 * ........................... Move to contour point 1
N100 H+120 * ........................................................... Move to contour point 2
N110 H+60 * ............................................................. Move to contour point 3
N120 G91 H–60 * ...................................................... Move to contour point 4, incremental value
N130 G90 H–60 * ...................................................... Move to contour point 5, absolute value
N140 H+240 * ........................................................... Move to contour point 6
N150 H+180 * ........................................................... Move to contour point 1
N160 G10 G40 R+70 H+170 * .................................. Retract the tool in X, Y; cancel radius compensation
N170 Z+100 M02 * .................................................... Retract the tool in Z
N9999 %S530I G71 *
```

Circular path G12/G13/G15 around pole I, J, K

The polar coordinate radius is also the radius of the arc. It is already defined by the distance from the POLE to the starting point Ⓢ.

Input

- Polar coordinate angle H for arc end point

- You can enter values from –5400° to +5400° for H.

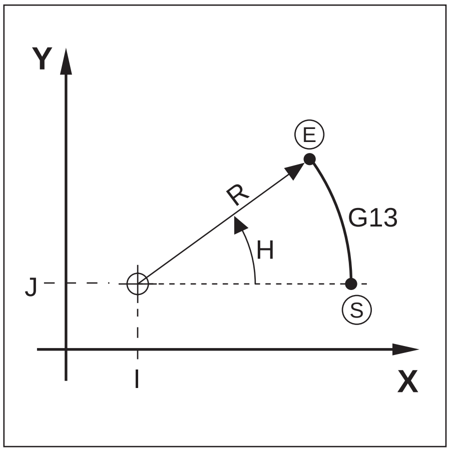


Fig. 5.36: Circular path around a pole

Defining the direction of rotation

You can program the following directions of rotation:

• Clockwise direction of rotation	G12
• Counterclockwise direction of rotation	G13
• No direction of rotation defined The tool moves in the direction of rotation defined in an earlier block.	G15

G 1 2	Program the circle with polar coordinates and clockwise rotation.
H 3 0 END □	Enter the angle of the arc end point, for example H = 30°. Terminate the block.

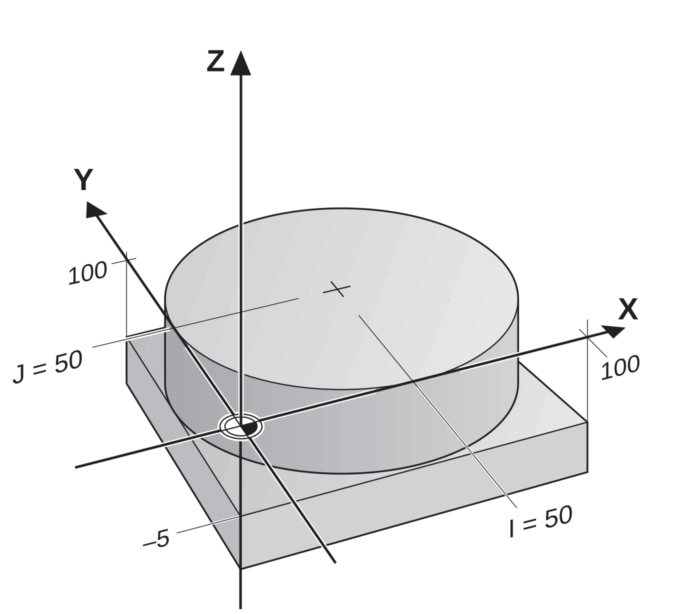
If necessary, enter also:

Radius compensation R
Feed rate F
Miscellaneous function M

*Resulting NC block: G12 H30 **

Example for exercise: Milling a full circle

Circle center coordinates:	X	=	50 mm
	Y	=	50 mm
Radius:	R	=	50 mm
Milling depth:	Z	=	– 5 mm
Tool radius:	R	=	15 mm



Part program

```
%S532I G71 * ................................ Begin program
N10 G30 G17 X+0 Y+0 Z–20 * .................. Define the workpiece blank
N20 G31 G90 X+100 Y+100 Z+0 *
N30 G99 T25 L+0 R+15 * ...................... Define the tool
N40 T25 G17 S1500 * ......................... Call the tool
N50 G00 G40 G90 Z+100 M06 * ................. Retract the spindle and insert the tool
N60 I+50 J+50 * ............................. Set the pole
N70 G10 R+70 H+280 * ........................ Pre-position in X, Y to polar coordinates
N80 Z–5 M03 * ............................... Pre-position to working depth
N90 G11 G41 R+50 H–90 F100 * ................ Move with radius compensation and reduced
                                              feed to the first contour point
N100 G26 R10 * .............................. Smooth (tangential) approach
N110 G12 H+270 * ............................ Circle to end point H = 270°, negative direction
                                           of rotation
N120 G27 R10 * .............................. Smooth (tangential) departure
N130 G10 G40 R+70 H–110 * ................... Retract tool in X, Y; cancel radius compensation
N140 Z+100 M02 * ............................ Retract tool in Z
N9999 %S532I G71 *
```

Circular path G16 with tangential connection

The tool moves on a circular path, starting tangentially (at ②) from a preceding contour element (① to ②).

Input:

- Polar coordinate angle H of the arc end point Ⓔ
- Polar coordinate radius R of the arc end point Ⓔ

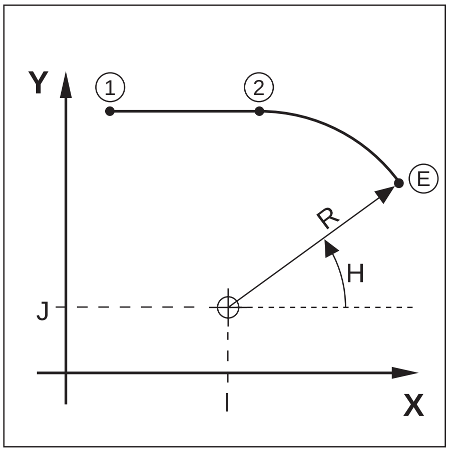


Fig. 5.37: Circular path around a pole, tangential connection

- The transition points must be defined exactly.
- The POLE is not the center of the contour arc.

G 1 6	Circle with polar coordinates and clockwise rotation.
R 1 0	Enter the distance from the pole to the arc end point, for example R= 10 mm.
H 8 0 END □	Enter the angle from the angle reference axis to R, for example H = 80°; terminate the block.

If necessary, enter also:

Radius compensation R
Feed rate F
Miscellaneous function M

*Resulting NC block: G16 R+10 H+80 **

Helical interpolation

A helix is the combination of a circular movement in a main plane and a linear movement perpendicular to the plane.

A helix is programmed only in polar coordinates.

Applications

You can use helical interpolation with form cutters to machine:

- Large-diameter internal and external threads
- Lubrication grooves

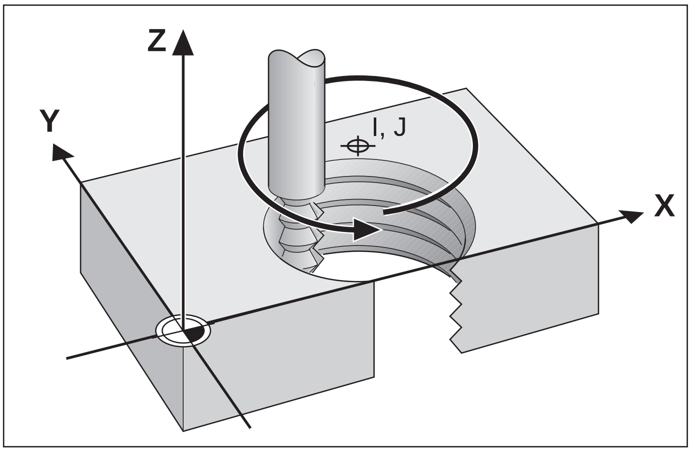


Fig. 5.38: Helix: a combination of circular and linear paths

Input

- Total incremental angle of tool traverse on the helix
- Total height of the helix

Input angle

Calculate the incremental polar coordinate angle G91 H as follows:
H = n · 360°.
n = number of revolutions of the helical path

For G91 H you can enter any value from –5400° to +5400° (n = 15).

Input height

Enter the helix height h in the tool axis. The height is calculated as:
h = n x P,
n = number of thread revolutions
P = thread pitch

Radius compensation

Enter the radius compensation for the helix according to the table at right.

Internal thread	Work direction	Rotation	Radius comp.
Right-hand	Z+	G13	G41
Left-hand	Z+	G12	G42
Right-hand	Z–	G12	G42
Left-hand	Z–	G13	G41

External thread	Work direction	Rotation	Radius comp.
Right-hand	Z+	G13	G42
Left-hand	Z+	G12	G41
Right-hand	Z–	G12	G41
Left-hand	Z–	G13	G42

Fig. 5.39: The shape of the helix determines the direction of rotation and the radius compensation

To program a helix:

G 1 2	Helix in clockwise direction of rotation.
G 9 1 H 1 0 8 0	Enter the total angle of tool traverse along the helix as an incremental value, for example H = 1080°.
Z 4 • 5 END □	Enter the total height of the helix in the tool plane as an incremental value, for example Z = 4.5 mm. Terminate the block.

If necessary, enter also:

Radius compensation
Feed rate F
Miscellaneous function M

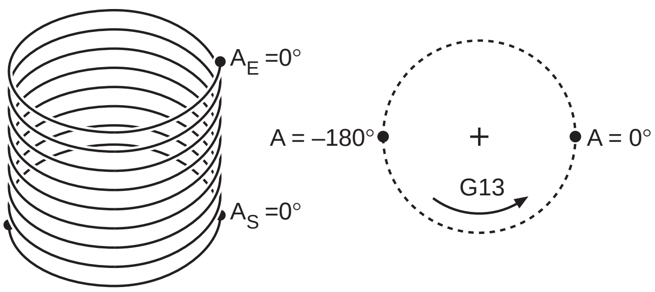
*Resulting NC block: G12 G91 H+1080 Z+4.5 **

Example for exercise: Tapping

Given data

Thread:

Right-hand internal thread	M64 x 1.5
Pitch P:	1.5 mm
Start angle A_S:	0°
End angle A_E:	360° = 0° at Z_E = 0
Thread revolutions n_T:	8
Thread overrun	
• at start of thread n_S:	0.5
• at end of thread n_E:	0.5
Number of cuts:	1



Calculating the input values

- Total height h: $H = P \cdot n$
  $P = 1.5$ mm
  $n = n_T + n_A + n_E = 9$
  $h = 13.5$ mm
- Incremental polar coordinate angle H: $H = n \cdot 360°$
  n = 9 (see total height H)
  IPA = 360° · 9 = 3240°
- Start angle A_S with thread overrun $n_{S:}$ $n_S = 0.5$

  The start angle of the helix is advanced by 180° (n = 1 corresponds to 360°). With positive rotation this means that
  A_S with n_S = A_S – 180° = –180°
- Starting coordinate: $Z = P \cdot (n_T + n_S)$
  $= -1.5 \cdot 8.5$ mm
  $= -12.75$ mm

  The thread is being cut in an upward direction towards $Z_E = 0$; therefore Z_S is negative.

Part program

```
%S536I G71 * ........................................... Begin program
N10 G30 G17 X+0 Y+0 Z–20 * ................. Define the workpiece blank
N20 G31 G90 X+100 Y+100 Z+0 *
N30 G99 T11 L+0 R+5 * ........................... Define the tool
N40 T11 G17 S2500 * ............................... Call the tool
N50 G00 G40 G90 Z+100 M06 * .............. Retract the spindle and insert the tool
N60 X+50 Y+30 * ..................................... Pre-position in the bore center in X, Y
N70 G29 * ................................................ Capture position as a pole
N80 Z–12 M03 * ....................................... Move the tool to starting depth
N90 G11 G41 R+32 H–180 F100 * ........... Move with radius compensation and reduced feed to the first contour
                                             point
N100 G13 G91 H+3240 Z+13.5 F200 *      Helical interpolation; incremental angle and tool movement in the Z axis
N110 G00 G40 G90 X+50 Y+30 * ............ Retract in X, Y(absolute values), cancel radius compensation
N120 Z+100 M02 * .................................. Retract in Z
N9999 %S536I G71 *
```

5.6 M Functions for Contouring Behavior and Coordinate Data

The following miscellaneous functions enable you to change the standard contouring behavior of the TNC in certain situations, such as:

- Smoothing corners
- Inserting transition arcs at non-tangential transitions between straight lines
- Machining small contour steps
- Machining open contour corners
- Entering machine-reference coordinates

Smoothing corners: M90

Standard behavior - without M90

At angular transitions such as internal corners and contours without radius compensation, the TNC stops the axes briefly.

Advantages:

- Reduced wear on the machine
- High definition of (outside) corners

Note:

In program blocks with radius compensation (G41/G42), the TNC automatically inserts a transition arc at external corners.

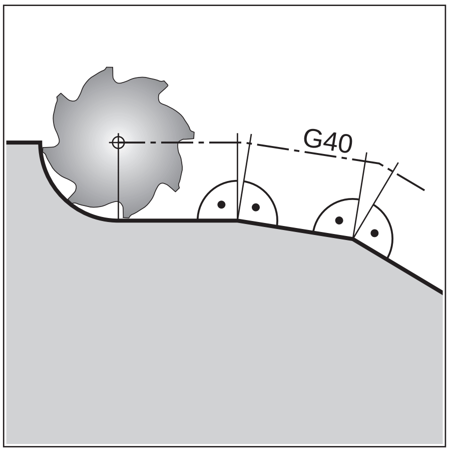


Fig. 5.40: Standard contouring behavior with G40 and without M90

Smoothing corners with M90

The tool moves around corners at constant speed.

Advantages:

- Provides a smoother, more continuous surface
- Reduces machining time

Application example:
Surfaces consisting of several straight line elements.

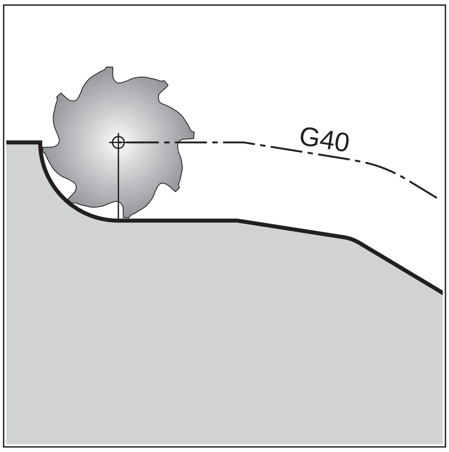


Fig. 5.41: Contouring behavior with G40 and M90.

Duration of effect

The miscellaneous function M90 is effective only in the blocks in which it is programmed. Operation with servo lag must be active.

A limit value can be set in machine parameter MP7460 (see page 11-9) below which the tool will move at constant feed rate (valid for operation both with servo lag and with feed precontrol). This value is valid regardless of M90.

Machining small contour steps: M97

Standard behavior – without M97

The TNC inserts a transition arc at outside corners. At very short contour steps this would cause the tool to damage the contour. In such cases the TNC interrupts the program run and displays the error message TOOL RADIUS TOO LARGE.

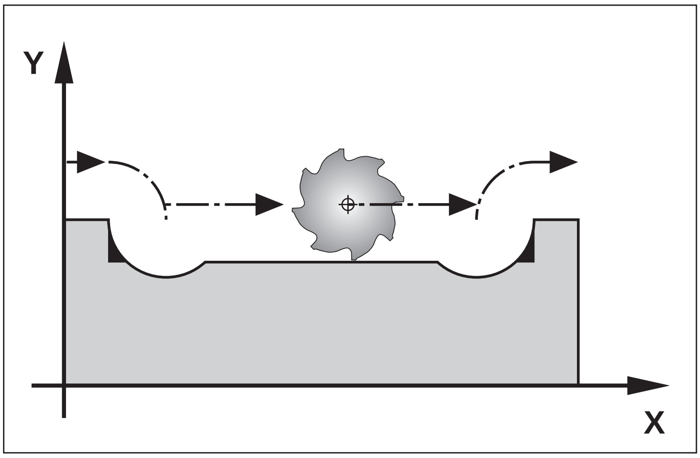


Fig. 5.42: Standard behavior without M97 if the block were to be executed as programmed

Machining contour steps – with M97

The TNC calculates the contour intersection Ⓢ (see figure) for the contour elements – as at inside corners – and moves the tool over this point. M97 is programmed in the same block as the outside corner point.

Duration of effect

The miscellaneous function M97 is effective only in the blocks in which it is programmed.

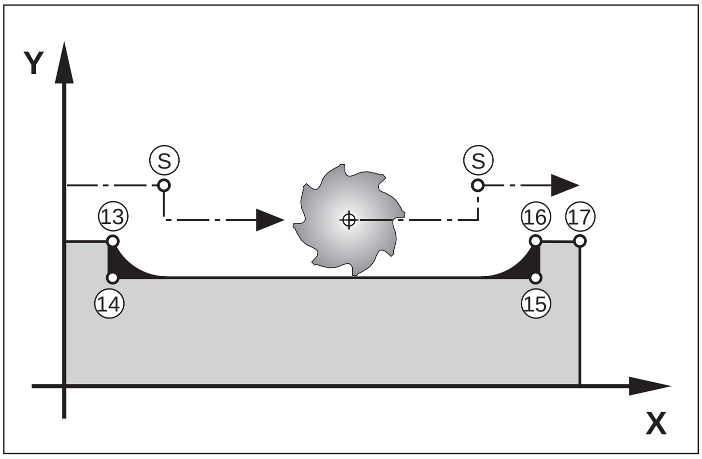


Fig. 5.43: Contouring behavior with M97

A contour machined with M97 is less complete than one without. You may wish to rework the contour with a smaller tool.

Program example

```
.
.
.
N5    G99 L ... R+20 ................................................ Large tool radius
.
.
.
N20   G01 X ... Y ... M97 .......................................... Move to contour point 13
N30   G91 Y–0.5 ....................................................... Machine the small contour step 13-14
N40   X+100 ............................................................. Move to contour point 15
N50   Y+0.5 M97 ...................................................... Machine the small contour step 15-16
N60   G90 X ... Y ... .................................................. Move to contour point 17
.
.
.
```

The outer corners are programmed in blocks N20 and N50: these are the blocks in which you program M97.

Machining open contours: M98

Standard behavior – without M98

The TNC calculates the intersections Ⓢ of the radius-compensated tool paths at inside corners and changes traverse direction at these points. If the corners are open on one side, however, machining is incomplete.

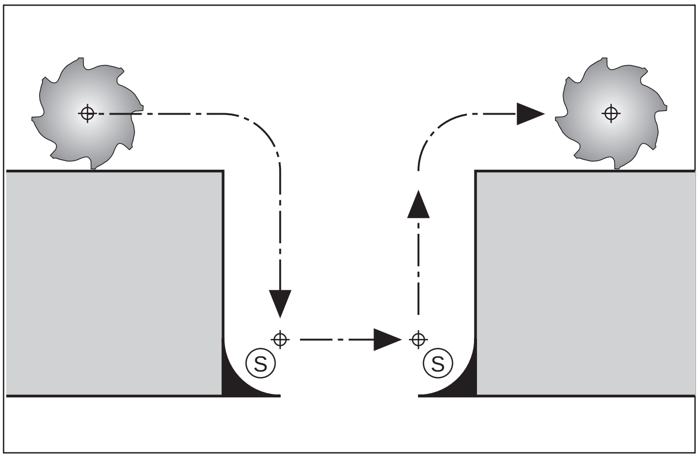


Fig. 5.44: Tool path without M98

Machining open corners – with M98

With the miscellaneous function M98 the TNC temporarily suspends radius compensation to ensure that both corners are completely machined.

Duration of effect

The miscellaneous function M98 is effective only in the blocks in which it is programmed.

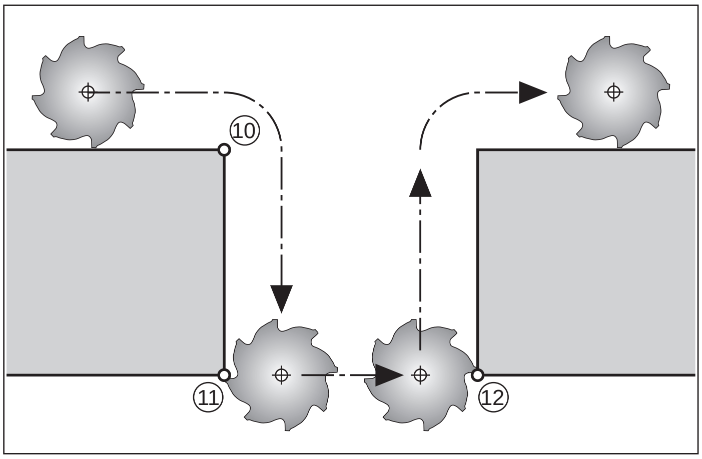


Fig. 5.45: Tool path with M98

Program example

```
 .
 .
 .
N10   X ... Y ... G41 F .. ........................................... Move to contour point 10
N20   X ... Y–... M98................................................ Machine contour point 11
N30   X + ... ............................................................. Move to contour point 12
 .
 .
 .
```

Programming machine-referenced coordinates: M91/M92

Standard behavior

Coordinates are referenced to the workpiece datum (see page 1-9).

Scale reference point

The position feedback scales are provided with one or more reference marks. Reference marks are used to indicate the position of the scale reference point. If the scale has only one reference mark, its position is the scale reference point. If the scale has several – distance-coded – reference marks, then the scale reference point is indicated by the left-most reference mark (at the beginning of the measuring range).

Machine datum – miscellaneous function M91

The machine datum is required for the following tasks:

- Defining the limits of traverse (software limit switches)
- Moving to machine-referenced positions (e.g. tool-change position)
- Setting the workpiece datum

The machine tool builder defines the distance for each axis from the scale reference point to the machine datum in a machine parameter.

If you want the coordinates in a positioning block to be referenced to the machine datum, end the block with the miscellaneous function M91.

Coordinates that are referenced to the machine datum are indicated in the display with REF.

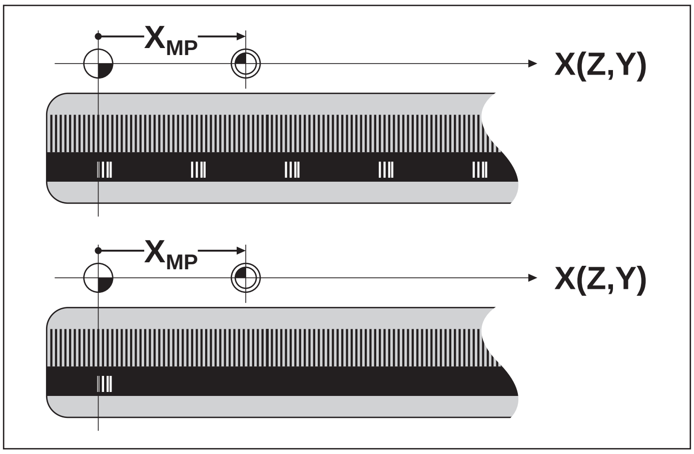


Fig. 5.46: Scale reference point ⌖ and machine datum ◎ for scales with one or several reference marks

Additional machine datum – miscellaneous function M92

Besides to the machine datum, the machine tool builder can define another machine-referenced position, the additional machine datum.

The machine tool builder defines the distance for each axis from the machine datum to the additional machine datum.

If you want the coordinates in a positioning block to be referenced to the additional machine datum, end the block with the miscellaneous function M92.

The values for radius compensation remain effective, even if you have programmed the coordinates with M91 of M92.

Workpiece datum

The user enters the coordinates of the datum for workpiece machining in the MANUAL OPERATION mode (see page 2-7).

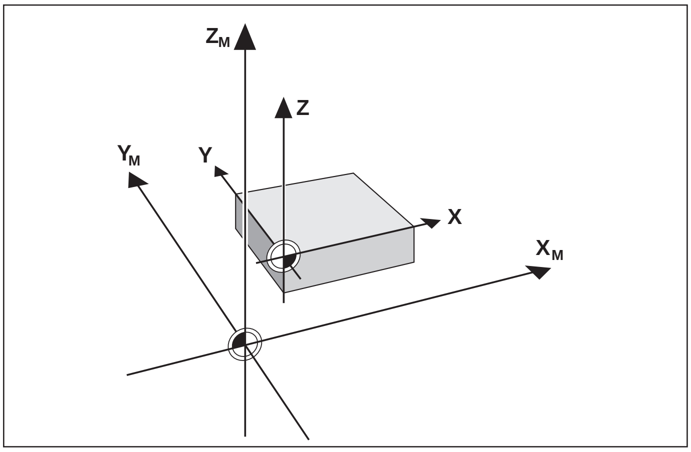


Fig. 5.47: Machine datum ⌖ and workpiece datum ⌖

5.7 Positioning with Manual Data Input (MDI)

In the POSITIONING WITH MANUAL DATA INPUT mode of operation you can use G07 to enter and execute single-axis positioning blocks. The entered positioning blocks are not stored in the TNC memory.

Application examples:

- Pre-positioning
- Face milling

POSITIONING MANUAL DATA INPUT

Select the MDI function.

G 7	Select a single axis positioning block.
G 9 0	For example: Select programming absolute values.
X 5	For example: Position in X at 5 mm.
F 1 5 0 0 END □	For example: Feed rate F=1500 mm/min; terminate the block.
I	Start the positioning block.

6 Subprograms and Program Section Repeats

Subprograms and program section repeats enable you to program a machining sequence once and then run it as often as you wish.

Labels

Subprograms and program section repeats are marked by labels.

A label carries a number from 0 to 254. Each label number (except 0) can only appear once in a program. Labels are assigned with the command G98.

LABEL 0 marks the end of a subprogram.

6.1 Subprograms

Principle

The (main) program is executed up to the block in which the subprogram is called with Ln,0 (①).

Then the subprogram is executed from beginning to end (G98 L0) (②).

Finally, the main program is resumed from the block after the subprogram call (③).

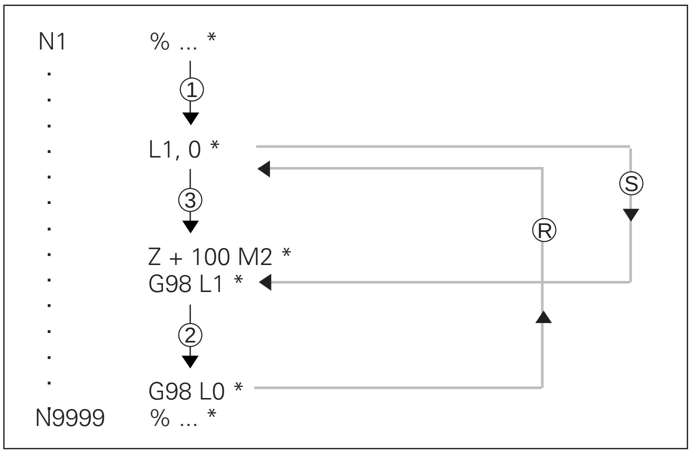


Fig. 6.1: Flow diagram for a subprogram; Ⓢ = jump, Ⓡ = return jump

Operating limits

- One main program can contain up to 254 subprograms.
- Subprograms can be called in any sequence and as often as desired.
- A subprogram cannot call itself.
- Subprograms should be located at the end of the main program (after the block with M2 or M30).
- If subprograms are located in the program before the block with M02 or M30, they will be executed at least once even without being called.

Programming and calling subprograms

To mark the beginning of the subprogram:

G 9 8 ENT	Select the label setting function.

LABEL NUMBER?	
e.g. 5 END □	In this example, the subprogram begins with LABEL 5.

*Resulting NC block: G98 L5 **

To mark the end of the subprogram:

A subprogram must always end with label number 0.

G 9 8 ENT	Select the label setting function.

LABEL NUMBER?	
0 END □	End of subprogram.

*Resulting NC block: G98 L0 **

To call the subprogram:

A subprogram is called with its label number.

L 5 . 0 END □	Calls the subprogram following LBL 5.

*Resulting NC block: L5,0 **

The command L0,0 is not allowed because label 0 can only be used to mark the end of a subprogram.

Example for exercise: Group of four holes at three different locations

The holes are drilled with cycle G83 PECKING. You enter the total hole depth, setup clearance, drilling feed rate, etc. once in the cycle. You can then call the cycle with the miscellaneous function M99 (see page 8-3).

Coordinates to the first hole in each group:

Group ①	X = 15 mm	Y = 10 mm
Group ②	X = 45 mm	Y = 60 mm
Group ③	X = 75 mm	Y = 10 mm

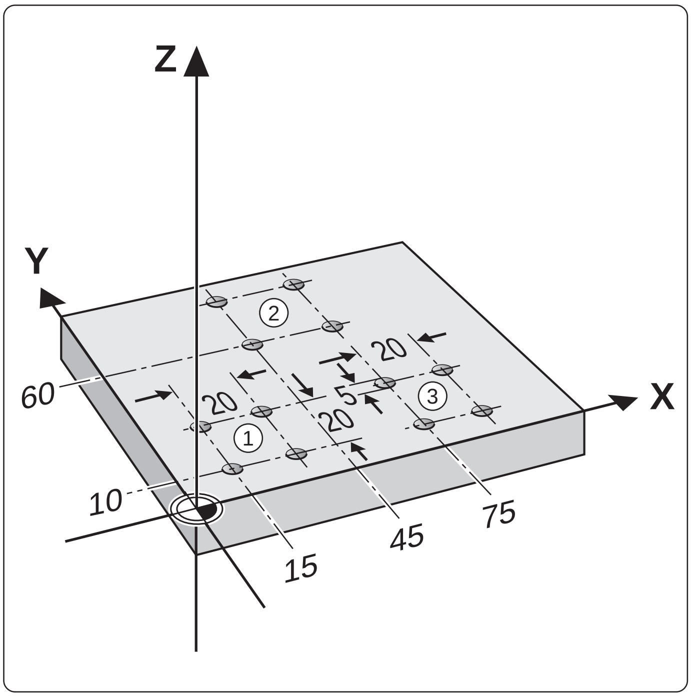
Spacing of holes:

X	=	20 mm
Y	=	20 mm

Total hole depth (DEPTH):

Z	=	10 mm

Hole diameter:Ø	=	5 mm



Part program

```
%S64I G71 * ...................................... Begin program
N10 G30 G17 X+0 Y+0 Z–20 * ....................... Define the workpiece blank
N20 G31 G90 X+100 Y+100 Z+0 *
N30 G99 T1 L+0 R+2.5 * ........................... Define the tool
N40 T1 G17 S3500 * ............................... Call the tool
N50 G83 P01 –2 P02 –10 P03 –5 P04 0
P05 100 * ........................................ Cycle definition PECKING (see page 8-5)
N60 G00 G40 G90 Z+100 M06 * ...................... Retract the spindle and insert the tool
N70 X+15 Y+10 * .................................. Move to hole group 1
N80 Z+2 M03 * .................................... Pre-position in the infeed axis
N90 L1,0 * ....................................... Subprogram call (with block N90 the subprogram  is
                                                   executed)
N100 X+45 Y+60 * ................................. Move to hole group 2
N110 L1,0 * ...................................... Subprogram call
N120 X+75 Y+10 * ................................. Move to hole group 3
N130 L1,0 * ...................................... Subprogram call
N140 Z+100 M02 * ................................. Retract tool;
                                                   End of main program (M2); the subprogram is
                                                   entered after M2
N150 G98 L1 * .................................... Beginning of subprogram
N160 G79 * ....................................... Execute pecking for the first hole
N170 G91 X+20 M99 * .............................. Move to incremental position for second hole and drill
N180 Y+20 M99 * .................................. Move to incremental position for third hole and drill
N190 X–20 G90 M99 * .............................. Move to incremental position for fourth hole and drill;
                                                   Switch to absolute coordinates (G90)
N200 G98 L0 * .................................... End of subprogram
N9999 %S64I G71 * ................................ End of program
```

6.2 Program Section Repeats

As with subprograms, program section repeats are marked with labels.

Principle

The program is executed up to the end of the labelled program section (block with Ln,m) (①, ②).

Then the program section between the called LABEL and the label call is repeated the number of times entered for m (③, ④).

After the last repetition, the program is resumed (⑤).

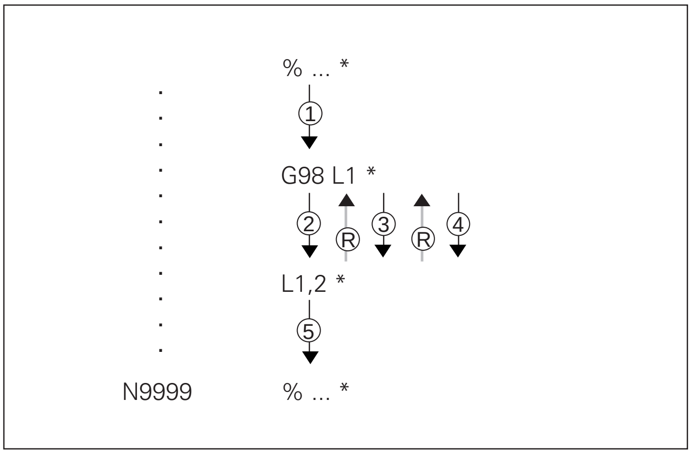


Fig. 6.2: Flow diagram with program section repeats;
Ⓡ = return jump

Programming notes

- A program section can be repeated up to 65 534 times in succession.
- The total number of times the program section will be carried out is always one more than the programmed number of repetitions.

Programming and calling a program section repeat

To mark the beginning:

G 9 8 ENT	Select the label setting function.

LABEL NUMBER?

e.g. 7 END □	Repeat the program section beginning with LABEL 7, for example.

*Resulting NC block: G98 L7 **

Number of repetitions

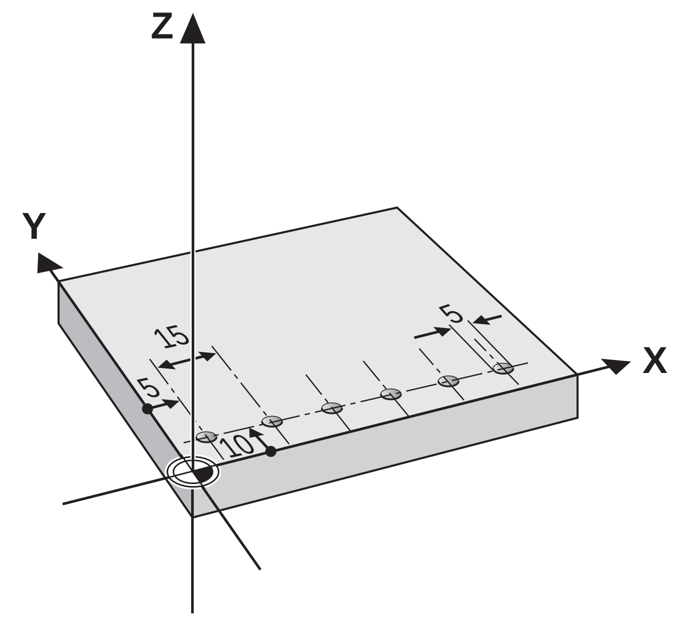
The number of repetitions is entered in the block which calls the label.
This block also identifies the end of the program section.

L 7 . 1 0 END □	Repeat the program section from LABEL 7 to this block 10 times. In this example, the program section will therefore be executed a total of 11 times.

*Resulting NC block: L7,10 **

Example for exercise: Row of holes parallel to X axis

Coordinates of first hole:	X	=	5 mm
	Y	=	10 mm
Spacing between holes:	IX	=	15 mm
Number of holes:	N	=	6
Total hole depth:	Z	=	10
Hole diameter:Ø	=	5 mm	



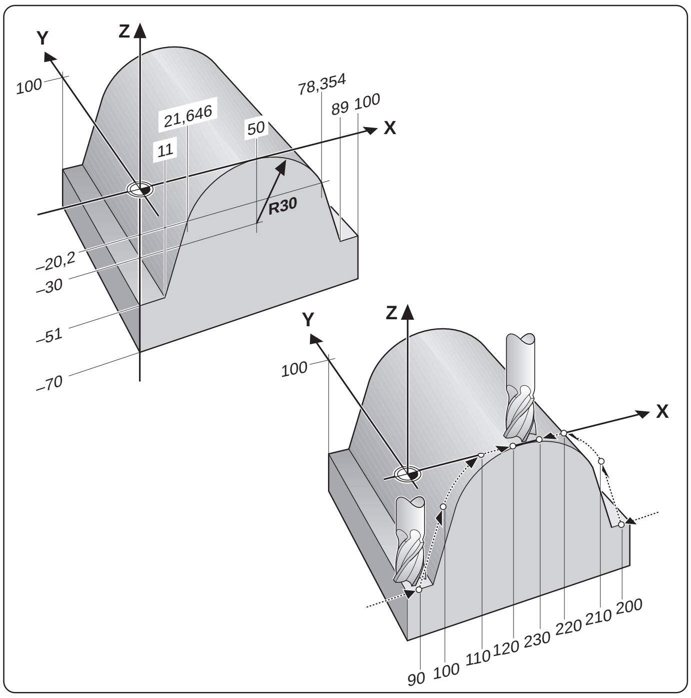
Part program

%S66I G71 *	Begin program
N10 G30 G17 X+0 Y+0 Z–20 *	Define the workpiece blank
N20 G31 G90 X+100 Y+100 Z+0 *	
N30 G99 T1 L+0 R+2.5 *	Define the tool
N40 T1 G17 S3500 *	Call the tool
N50 G00 G40 G90 Z+100 M06 *	Retract the spindle and insert the tool
N60 X–10 Y+10 Z+2 M03 *	Pre-position in the negative X direction
N70 G98 L1 *	Beginning of program section to be repeated
N80 G91 X+15 *	Move to incremental hole position
N90 G01 G90 Z–10 F100 *	Drill (absolute value)
N100 G00 Z+2 *	Retract the tool
N110 L1,5 *	Call LABEL 1; repeat program section between blocks N70 and N110 five times (for six holes)
N120 Z+100 M02 *	Retract the tool in Z
N9999 %S66I G71 *	

Example for exercise: Milling with program section repeat without radius compensation

Machining sequence

- Upward milling direction
- Machine the area from X = 0 to 50 mm (program all X coordinates with the tool radius subtracted) and from Y = 0 to 100 mm : G98 L1
- Machine the area from X = 50 to 100 mm (program all X coordinates with the tool radius added) and from Y = 0 to 100 mm : G98 L2
- After each upward pass, the tool is moved by an increment of +2.5 mm in the Y axis.



The illustration to the right shows the block numbers containing the end points of the corresponding contour elements.

Part program

```
%S67I G71 * ............................................................. Begin program
N10 G30 G17 X+0 Y+0 Z–70 * ................................ Define workpiece blank (note: blank form has changed)
N20 G31 G90 X+100 Y+100 Z+0 *
N30 G99 T1 L+0 R+10 * .......................................... Define the tool
N40 T1 G17 S1750 * ................................................ Call the tool
N50 G00 G40 G90 Z+100 M06 * .............................. Retract the spindle and insert the tool
N60 X–20 Y–1 M03 * ............................................... Pre-position in the X, Y plane
N70 G98 L1 * ........................................................... Begin program section 1
N80 G90 Z–51 *
N90 G01 X+1 F100 *
N100 X+11.646 Z–20.2 * ......................................... Program section for machining from
N110 G06 X+40 Z+0 * .............................................. X = 0 to 50 mm and Y = 0 to 100 mm
N120 G01 X+41 *
N130 G00 Z+10 *
N140 X–20 G91 Y+2.5 *
N150 L1,40 * ............................................................ Call LABEL 1, repeat program section between blocks
                                                                          N70 and N150 40 times
N160 G90 Z+20 * ..................................................... Retract the tool
N170 X+120 Y–1 * ................................................... Pre-position for program section 2
N180 G98 L2 * .......................................................... Beginning of program section 2
N190 G90 Z–51 *
N200 G01 X+99 F100 *
N210 X+88.354 Z–20.2 * ......................................... Program section for machining from
N220 G06 X+60 Z+0 * .............................................. X = 50 to 100 mm and Y = 0 to 100 mm
N230 G01 X+59 *
N240 G00 Z+10 *
N250 X+120 G91 Y+2.5 *
N260 L2,40 * ............................................................ Call LABEL 2, repeat program section between blocks
                                                                          N180 and N260 40 times
N270 G90 Z+100 M02 * .......................................... Retract the tool
N9999 %S67I G71 *
```

6.3 Main Program as Subprogram

Principle

A program is executed until another program is called (block with %) (①).

The called program is executed from beginning to end (②).

Execution of the program from which the other program was called is then resumed with the block following the program call (③).

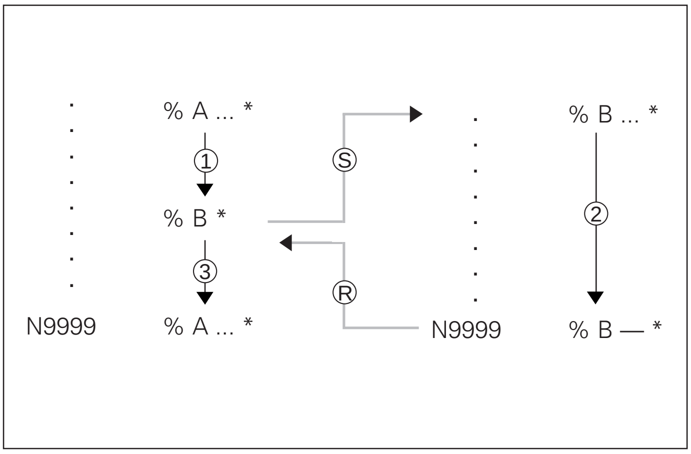


Fig. 6.3: Flow diagram of a main program as subprogram; Ⓢ = jump, Ⓡ = return jump

Operating limits

- Programs called from an external data storage medium must not contain any subprograms or program section repeats.
- No labels are needed to call main programs as subprograms.
- The called program must not contain the miscellaneous functions M2 or M30.
- The called program must not contain a jump into the calling program.

To call a main program as a subprogram

% ▶ **PROGRAM NAME?**

Enter the main program call and the name of the program you want to call.

Resulting NC block: % NAME

A main program can also be called with cycle G39 (see page 8-38).

6.4 Nesting

Subprograms and program section repeats can be nested in the following variations:

- Subprograms in subprograms
- Program section repeats in program section repeats
- Subprograms can be repeated
- Program section repeats can appear in subprograms

Nesting depth

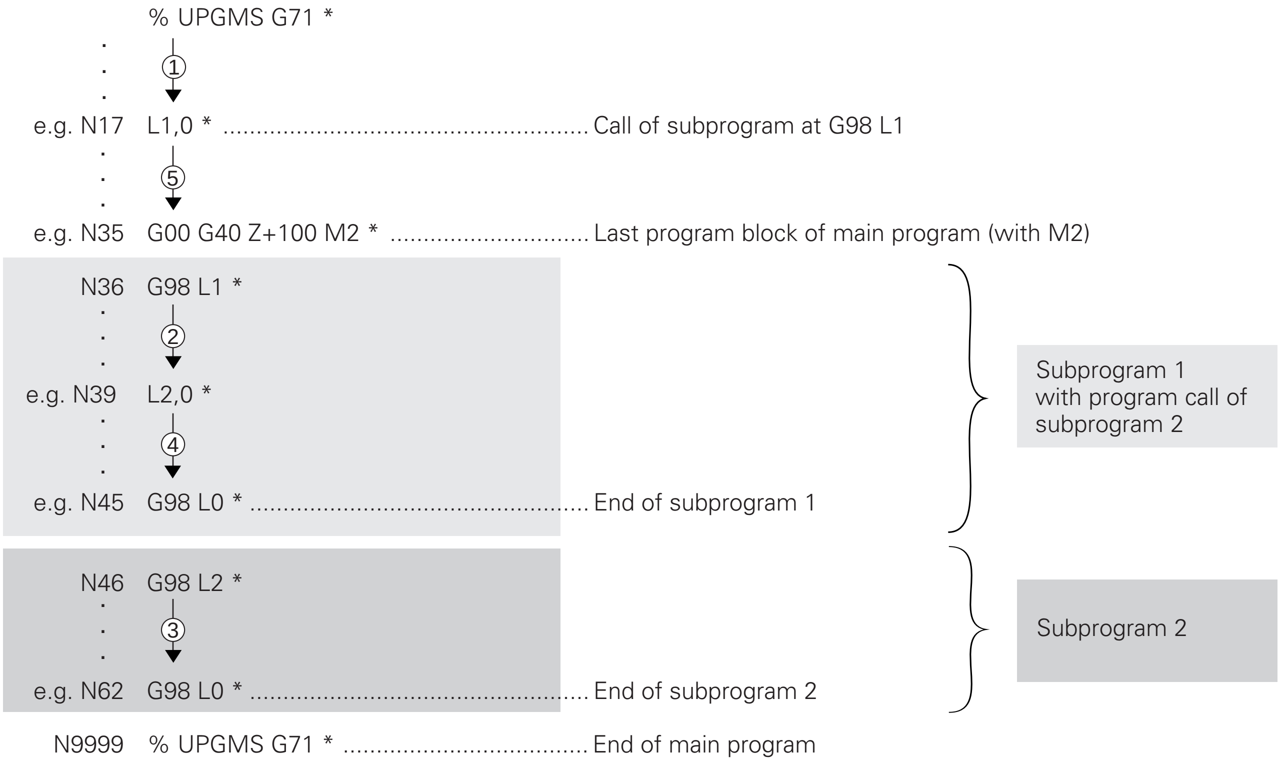
The nesting depth is the number of successive levels for which subprograms or program sections can call further subprograms or program section repeats.

Maximum nesting depth for subprograms: 8

Maximum nesting depth for calling main programs: 4

Subprogram in a subprogram

Program layout



```
                % UPGMS G71 *
          .
          .       (1)
          .
e.g. N17  L1,0 * ........................ Call of subprogram at G98 L1
          .
          .       (5)
          .
e.g. N35  G00 G40 Z+100 M2 * ............ Last program block of main program (with M2)
     N36  G98 L1 *
          .
          .       (2)
          .
e.g. N39  L2,0 *                                          Subprogram 1
          .                                               with program call of
          .       (4)                                     subprogram 2
          .
e.g. N45  G98 L0 * ...................... End of subprogram 1
     N46  G98 L2 *
          .
          .       (3)                                     Subprogram 2
          .
e.g. N62  G98 L0 * ...................... End of subprogram 2
   N9999  % UPGMS G71 * ................. End of main program
```

Sequence of program execution

Step 1: Main program UPGMS is executed up to block 17.

Step 2: Subprogram 1 is called and executed up to block 39.

Step 3: Subprogram 2 is called and executed up to block 62.
End of subprogram 2 and return to the subprogram from which it was called.

Step 4: Subprogram 1 is executed from block 40 to block 45.
End of subprogram 1 and return to main program UPGMS.

Step 5: Main program UPGMS is executed from block 18 to block 35.
Return jump to block 1 and program end.

Example for exercise: Group of four holes at three positions (see page 6-4), but with three different tools

Machining sequence:
Countersinking - Pecking - Tapping

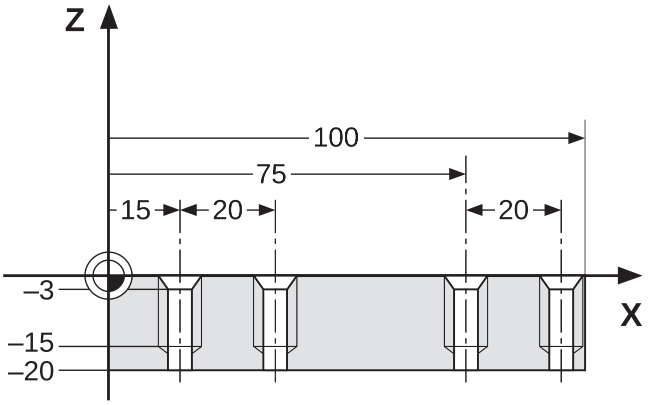
The drilling operation is programmed with cycle G83: PECKING (see page 8-4) and cycle G84: TAPPING (see page 8-6). The groups of holes are approached in one subprogram, and the machining is performed in a second subprogram.

Coordinates of the first hole in each group:

①	X = 15 mm	Y = 10 mm
②	X = 45 mm	Y = 60 mm
③	X = 75 mm	Y = 10 mm
Spacing between holes	IX = 20 mm	IY = 20 mm

Hole data:

Countersinking	ZC = 3 mm	Ø = 7 mm
Pecking	ZP = 15 mm	Ø = 5 mm
Tapping	ZT = 10 mm	Ø = 6 mm

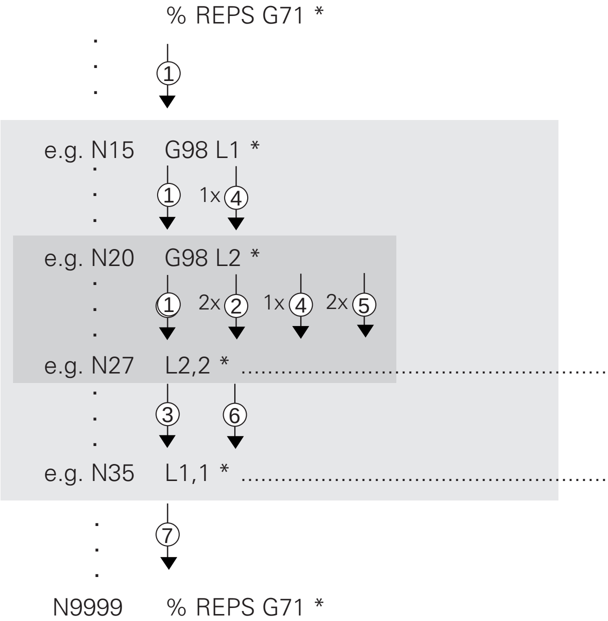


Part program

```
%S610I G71 * .......................................................... Begin program
N10 G30 G17 X+0 Y+0 Z–20 * ................................ Define the workpiece blank
N20 G31 G90 X+100 Y+100 Z+0 *
N30 G99 T25 L+0 R+2.5 * ....................................... Tool definition for pecking
N40 G99 T30 L+0 R+3 * .......................................... Tool definition for countersinking
N50 G99 T35 L+0 R+3.5 * ....................................... Tool definition for tapping
N60 T30 G17 S3000 * ............................................... Tool call for countersinking
N70 G83 P01 –2 P02 –3 P03 –3 P04 0
P05 100 * ................................................................. Cycle definition for pecking
N80 L1,0 * ............................................................... Call of subprogram 1
N90 T25 G17 S2500 * ............................................... Tool call for pecking
N100 G83 P01 –2 P02 –25 P03 –10 P04 0
P05 150 * ................................................................. Cycle definition for pecking
N110 L1,0 * ............................................................. Call of subprogram 1
N120 T35 G17 S100 * ............................................... Tool call for tapping
N130 G84 P01 –2 P02 –15 P03 0.1 P04 100 * ......... Cycle definition for tapping
N140 L1,0 * ............................................................. Call of subprogram 1
N150 Z+100 M02 * .................................................. Retract the tool; end of main program
N160 G98 L1 * ......................................................... Beginning of subprogram 1
N170 G00 G40 G90 X+15 Y+10 M03 * .................... Move to hole group 1
N180 Z+2 * .............................................................. Pre-position in the infeed axis
N190 L2,0 * ............................................................. Call subprogram 2
N200 X+45 Y+60 * .................................................. Move to hole group 2
N210 L2,0 * ............................................................. Call subprogram 2
N220 X+75 Y+10 * .................................................. Move to hole group 3
N230 L2,0 * ............................................................. Call subprogram 2
N240 G98 L0 * ......................................................... End of subprogram 1
N250 G98 L2 * ......................................................... Beginning of subprogram 2
N260 G79 *
N270 G91 X+20 M99 * ............................................. Machine holes by sequentially activating the three cycles
N280 Y+20 M99 *
N290 X–20 G90 M99 *
N300 G98 L0 * ......................................................... End of subprogram 2
N9999 %S610I G71 *
```

Repeating program section repeats

Program layout



Program section between this block and G98 L2 (block 20) is repeated twice

Program section between this block and G98 L1 (block 15) is repeated once

Sequence of program execution

Step 1: Main program REPS is executed up to block 27.

Step 2: Program section between block 27 and block 20 is repeated twice.

Step 3: Main program REPS is executed from block 28 to block 35.

Step 4: Program section between block 35 and block 15 is repeated once.

Step 5: Repetition of step 2 within step ④.

Step 6: Repetition of step 3 within step ④.

Step 7: Main program REPS is executed from block 36 to block 50. End of program.

Repeating subprograms

Program layout

```
                 % UPGREP G71 *
         .                |
         .               ①
         .                ↓
e.g.     N10     G98 L1 *                |
         N11     L2,0 * ...........2x ③.... Subprogram call
         N12     L1,2 * ...............↓... Program section repeat
         .          |
         .         ④
         .          ↓
e.g.     N19     G00 G40 Z+100 M2 * ....... Last program block of main program with M2
         N20     G98 L2 * ................. Beginning of subprogram
         .          |
         .         ②
         .          ↓
e.g.     N28     G98 L0 * ................. End of subprogram
        N9999    % UPGREP G71 * ........... End of main program
```

Sequence of program execution

Step 1: Main program UPGREP is executed to block 11.

Step 2: Subprogram 2 is called and executed.

Step 3: Program section between block 12 and block 10 is repeated twice: subprogram 2 is repeated twice.

Step 4: Main program UPGREP is executed from block 13 to block 19. End of program.

Q Parameters are used for:

- **Programming families of parts**
- **Defining contours through mathematical functions**

A **family of parts** can be programmed in the TNC in a **single part program**. You do this by entering variables — called Q parameters — instead of numerical values.

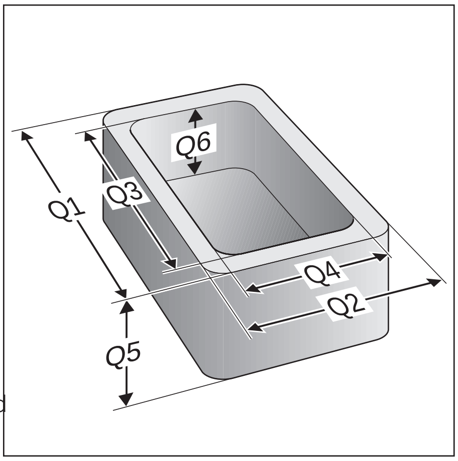


Fig. 7.1: Q parameters as variables

Q parameters can represent for example:

- Coordinate values
- Feed rates
- Spindle speeds
- Cycle data

A Q parameter is designated by the letter Q and a number between 0 and 123.

Q parameters also enable you to program **contours** that are defined through **mathematical functions**.

With Q parameters you can make the execution of machining steps dependent on **logical conditions**.

Q parameters and numerical values can also be **mixed** within a program.

The TNC automatically assigns data to some Q parameters. For example, parameter Q108 is assigned the current tool radius. You will find a list of these parameters in Chapter 11.

7.1 Part Families — Q Parameters Instead of Numerical Values

The Q parameter function D0: ASSIGN is used for assigning numerical values to Q parameters.
Example: N10 D00 Q10 P01+25 *

This enables you to enter variable Q parameters in the program instead of numerical values.
Example: G00 G40 G90 X + Q10 (corresponds to X + 25)

For part families, the characteristic workpiece dimensions can be programmed as Q parameters. Each of these parameters is then assigned a different value when the parts are machined.

Example

Cylinder with Q parameters

Cylinder radius	R	=	Q1
Cylinder height	H	=	Q2
Cylinder Z1:	Q1	=	+30
	Q2	=	+10
Cylinder Z2:	Q1	=	+10
	Q2	=	+50

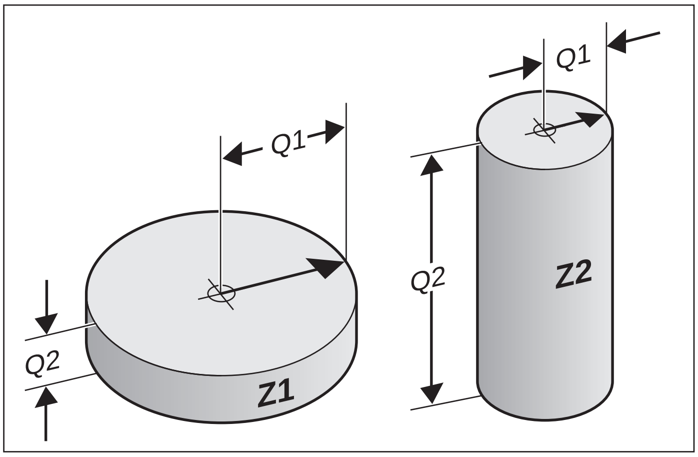


Fig. 7.2: Workpiece dimensions as Q parameters

To assign numerical values to Q parameters:

D 0 ENT	Select function D0: ASSIGN.

PARAMETER NUMBER FOR RESULT?

e.g. 5 ENT	Enter Q parameter number.

FIRST VALUE / PARAMETER?

e.g. 6 ENT	Enter value or another Q parameter whose value is to be assigned to Q5.

*Resulting NC block: N20 D00 Q05 P01 +6 **

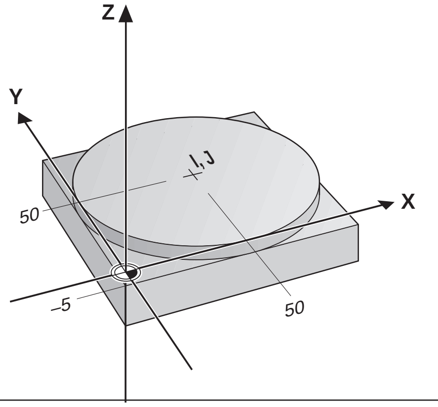
Example for exercise: Full circle

Circle center I,J:
X = 50 mm Y = 50 mm

Beginning and end of the circular arc:
X = 50 mm Y = 0 mm

Milling depth: ZM = –5 mm

Tool radius: R = 15 mm



Part program without Q parameters

```
%S520I G71 * ........................................................ Start of program
N10 G30 G17 X+1 Y+1 Z-20 * .................................. Definition of blank form MIN point
N20 G30 G90 X+100 Y+100 Z+0 * ........................... Definition of blank form MAX point
N30 G99 T6 L+0 R+15 * ......................................... Tool definition
N40 T6 G17 S500 * ............................................... Tool call
N50 I+50 J+50 * ................................................... Coordinates of the circle center
N60 G00 G40 G90 Z+100 M06 * .............................. Retract the spindle and insert the tool
N70 X+30 Y–20 * ................................................... Pre-position the tool
N80 Z–5 M03 * ...................................................... Pre-position the tool to working depth
N90 G01 G41 X+50 Y+0 F100 * ............................... Move to first contour point with radius compensation
N100 G02 X+50 Y+0 * ............................................. Mill circular arc around circle center I,J; coordinates of end
                                                                  point X = +50 and Y = 0; positive direction of rotation (G02)
N110 G00 G40 X+70 Y–20 * ..................................... Retract the tool in X, Y; cancel radius compensation
N120 Z+100 M02 * .................................................. Retract the tool in Z
N9999 %S520I G71 *
```

Part program with Q parameters

```
%3600741 G71 *
N10 D00 Q01 P01 +100 * ......................................... Clearance height
N20 D00 Q02 P01 +30 * ........................................... Start pos. X
N30 D00 Q03 P01 –20 * ........................................... Start-End pos. Y
N40 D00 Q04 P01 +70 * ........................................... End pos. X
N50 D00 Q05 P01 –5 * ............................................. Milling depth
N60 D00 Q06 P01+50 * ............................................ Circle center X
N70 D00 Q07 P01 +50 * ........................................... Circle center Y
N80 D00 Q08 P01 +50 * ........................................... Circle start point X
N90 D00 Q09 P01 +0 * ............................................. Circle start point Y
N100 D00 Q10 P01 +0 * ........................................... Tool length L
N110 D00 Q11 P01 +15 * ......................................... Tool radius R
N120 D00 Q20 P01 +100 * ....................................... Milling feed rate F
```

Blocks N10 to N120:
Assign numerical values to the Q parameters

```
N130 G30 G17 X+0 Y+0 Z–20 *
N140 G31 G90 X+100 Y+100 Z+0 *
N150 G99 T1 L+Q10 R+Q11 *
N160 T1 G17 S500 *
N170 I+Q6 J+Q7 *
N180 G00 G40 G90 Z+Q1 M06 *
N190 X+Q2 Y+Q3 *
N200 Z+Q5 M03 *
N210 G01 G41 X+Q8 Y+Q9 FQ20 *
N220 G02 X+Q8 Y+Q9 *
N230 G01 G40 X+Q4 Y+Q3 *
N240 Z+Q1 M02 *
N9999 %3600741 G71 *
```

Blocks N130 to N240:
Corresponding to blocks N10 to N120 from program S520I

7.2 Describing Contours Through Mathematical Functions

Overview

The mathematical functions assign the results of one of the following operations to a Q parameter:

D00: ASSIGN
e.g. N10 D00 Q05 P01 +60 *
Assigns a value directly

D01: ADDITION
e.g. N10 D01 Q01 P01 –Q2 P02 –5 *
Calculates and assigns the sum of two values

D02: SUBTRACTION
e.g. N10 D02 Q01 P01 +10 P02 +5 *
Calculates and assigns the difference between two values

D03: MULTIPLICATION
e.g. N10 D03 Q02 P01 +3 P02 +3 *
Calculates and assigns the product of two values

D04: DIVISION
e.g. N10 D04 Q04 P01 +8 P02 +Q02 *
Calculates and assigns the quotient of two values

Note: Division by 0 is not possible!

D05: SQUARE ROOT
e.g. N10 D05 Q20 P01 +4 *
Calculates and assigns the square root of a number

Note: Square root of a negative number is not possible!

The "values" in the overview above can be:

- two numbers
- two Q parameters
- a number and a Q parameter

The Q parameters and numerical values in the equations can be entered with positive or negative signs.

Programming example for fundamental operations

Assign the value 10 to parameter Q5, and assign the product of Q5 and 7 to parameter Q12.

D 0 ENT	Select Q parameter function D00 (ASSIGN).

PARAMETER NUMBER FOR RESULT ?

5 ENT	Enter parameter number, for example 5, and confirm.

FIRST VALUE / PARAMETER ?

1 0 END □	Assign numerical value to Q5, terminate block.

D 3 ENT	Select Q parameter function D03 (MULTIPLICATION).

PARAMETER NUMBER FOR RESULT?

1 2 ENT	Enter parameter number, for example Q12, and confirm.

FIRST VALUE OR PARAMETER?

Q 5 ENT	Enter Q5 (=10) and confirm.

SECOND VALUE OR PARAMETER?

7 END □	Enter 7, terminate block.

Resulting NC blocks: *N20 D00 Q05 P01 +10 **
*N30 D03 Q12 P01 +Q5 P02 +7 **

7.3 Trigonometric Functions

Sine, cosine and tangent are the terms for the ratios of the sides of right triangles. Trigonometric functions simplify many calculations.

For a right triangle,

Sine: $\sin \alpha = a / c$

Cosine: $\cos \alpha = b / c$

Tangent: $\tan \alpha = a / b = \sin \alpha / \cos \alpha$

Where

- c is the side opposite the right angle
- a is the side opposite the angle α
- b is the third side

The angle can be derived from the tangent:

$\alpha = \arctan \alpha = \arctan (a / b) = \arctan (\sin \alpha / \cos \alpha)$

Example: $a = 10$ mm
$b = 10$ mm
$\alpha = \arctan (a / b) = \arctan 1 = 45°$

Furthermore: $a^2 + b^2 = c^2$ $(a^2 = a \cdot a)$
$c = \sqrt{a^2 + b^2}$

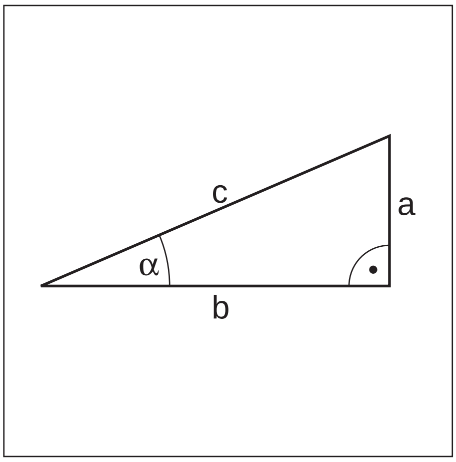


Fig. 7.3: Sides and angles on a right triangle

Overview

D06: SINE
e.g. N10 D06 Q20 P01 –Q05 *
Calculate the sine of an angle in degrees (°) and assign it to a parameter

D07: COSINE
e.g. N10 D07 Q21 P01 –Q05 *
Calculate the cosine of an angle in degrees (°) and assign it to a parameter

D08: ROOT SUM OF SQUARES
e.g. N10 D08 Q10 P01 +5 P02 +4 *
Take the square root of the sum of two squares, and assign it to a parameter

D13: ANGLE
e.g. N10 D13 Q20 P01 +10 P02 –Q01 *
Calculate the angle from the arc tangent of two sides or from the sine and cosine of the angle, and assign it to a parameter

7.4 If-Then Operations with Q Parameters

If-Then conditional operations enable the TNC to compare a Q parameter with another Q parameter or with a numerical value.

Jumps

The jump target is specified in the block through a label number. If the programmed condition is true, the TNC continues the program at the specified label; if it is false, the next block is executed.

To jump to another program, you enter a program call after the block with the target label (see page 6-8).

Overview

D09: IF EQUAL, JUMP

e.g. N10 D09 P01 +Q01 P02 +Q03 P03 5 *

If the two values or parameters are equal, jump to the specified label (here label 5).

D10: IF NOT EQUAL, JUMP

e.g. N10 D10 P01 +10 P02 –Q05 P03 10 *

If the two values or parameters are not equal, jump to the specified label (here label 19).

D11: IF GREATER THAN, JUMP

e.g. N10 D11 P01 +Q01 P02 –10 P03 5 *

If the first value or parameter is greater than the second value or parameter, jump to the specified label (here label 5).

D12: IF LESS THAN, JUMP

e.g. N10 D12 P01 +Q05 P02 +0 P03 1 *

If the first value or parameter is less than the second value or parameter, jump to the specified label (here label 1).

Unconditional jumps

Unconditional jumps are jumps which are always executed because the condition is always true.
Example:

N20 D09 P01 +10 P02 +10 P03 1 *

Program example

When Q5 becomes negative, a jump to program 100 will occur.

```
.
.
.
N50 D00 Q05 P01 +10 * ........................................... Assign value, for example 10, to parameter Q5
.
.
.
N90 D02 Q05 P01 +Q5 P02 +12 * ............................ Reduce the value of Q5
N100 D12 P01 +Q5 P02 +0 P03 5 * .......................... If +Q5 is less than 0, jump to label 5
.
.
.
N150 G98 L5 * ........................................................... Label 5
N160 %100 * ............................................................. Jump to program 100
.
.
.
```

7.5 Checking and Changing Q Parameters

Q parameters can be checked during program run or during a test run, and changed if necessary.

Preparation:

- A running program must be aborted (e.g. press machine STOP button and STOP key).
- If you are doing a test run, you must interrupt it.

To call a Q parameter:

Q ▶ **Q ▭ =**

e.g. 1 0 ENT	Select desired Q parameter (in this example Q10).

Q10 = + 100

The TNC displays the current value (in this example Q10 = 100).

e.g. 0 ENT	Change Q parameter (in this example Q10 is changed to 0).
ENT	Leave the Q parameter unchanged.

7.6 Output of Q Parameters and Messages

Displaying error messages

With the function D14: ERROR NUMBER you can call messages that were pre-programmed by the machine tool builder.

If the TNC encounters a block with D14 during a program run or test run, it interrupts the run and displays an error message. The program must then be restarted.

Input example:

N50 D14 P01 254 *

The TNC will display the text of error number 254.

Error number to be entered	Prepared dialog text
0 to 299	ERROR 0 to ERROR 299
300 to 399	PLC ERROR 01 to PLC ERROR 99
400 to 483	DIALOG 1 to 83
484 to 499	USER PARAMETER 15 to 0

The machine tool builder may have programmed a text that differs from the above.

Output through an external data interface

The function D15: PRINT transmits the values of Q parameters and error messages over the data interface. This enables you to send such data to external devices, for example to a printer.

- D15: PRINT with numerical values up to 254
  Example: N100 D15 P01 20 *
  Transmits the corresponding error message (see overview for D14).
- D15: PRINT with Q parameter
  Example: N200 D15 P01 Q20 *
  Transmits the value of the corresponding Q parameter.

Up to six Q parameters and numerical values can be transmitted simultaneously.
Example: N250 D15 P01 4 P02 Q05 P03 4 P04 Q25 *

Assigning values for the PLC

Function D19: PLC transmits up to two numerical values or Q parameters to the PLC.

Input increment and unit of measure: 1µm or 0.001°

Example: N25 D19 P01 +10 P02 +Q3 *

The number 10 corresponds to 10 µm or 0.01°

7.7 Measuring with the 3D Touch Probe During Program Run

The 3D touch probe can measure positions on a workpiece during program run.

Applications:

- Measuring differences in the height of cast surfaces
- Checking tolerances during machining

Enter G55 to activate the touch probe.
The touch probe is automatically pre-positioned (with rapid traverse from MP6150) and probes the specified position (with feed rate from MP6120).
The coordinate measured for the probe point is stored in a Q parameter.

The TNC interrupts the probing process if the probe is not deflected within a certain range (range selected with MP6130).

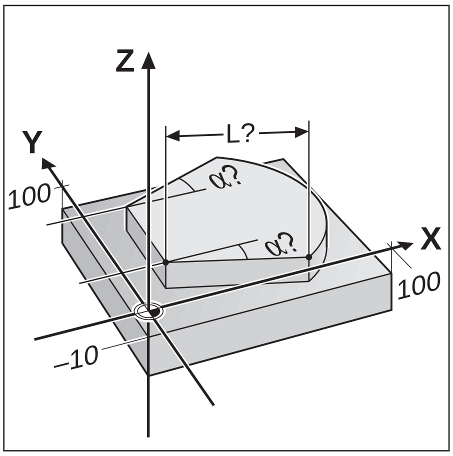


Fig. 7.4: Workpiece dimensions to be measured

To program the use of a touch probe:

G 5 5 ENT	Select the touch probe function.
PARAMETER NUMBER FOR RESULT ?	
e.g. 5 ENT	Enter the number of the Q parameter to which the coordinate is to be assigned, for example Q5.
PROBING AXIS/PROBING DIRECTION?	
e.g. X e.g. +/–	Enter the probing axis for the coordinate, for example X. Select and confirm the probing direction.
e.g. X 5 e.g. Y 0 e.g. Z +/– 5	Enter all coordinates of the pre-positioning point values, for example X = 5 mm, Y = 0, Z = –5 mm.
END □	Conclude input.

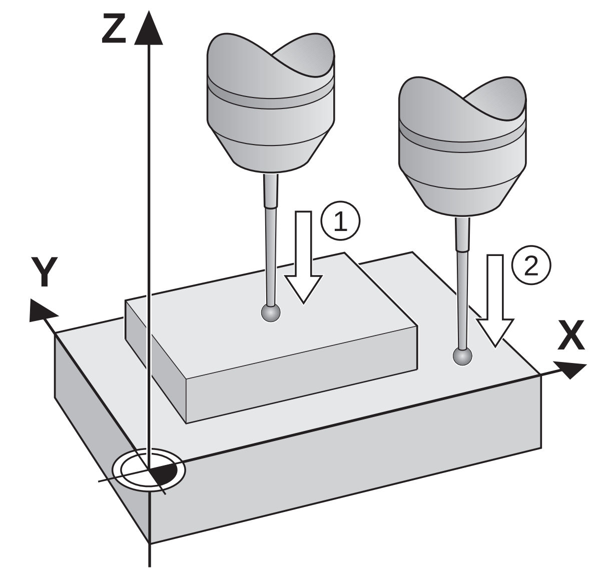
*Resulting NC block: N150 G55 P01 05 P02 X- X+5 Y+0 Z-5 **

Pre-position the touch probe manually such that it will not collide with the workpiece when it moves toward the programmed position.

Example for exercise: Measuring the height of an island on a workpiece

Coordinates for pre-positioning the
3D touch probe

Touch point 1:	X	=	+ 20	mm (Q11)
	Y	=	50	mm (Q12)
	Z	=	10	mm (Q13)
Touch point 2:	X	=	+ 50	mm (Q21)
	Y	=	10	mm (Q22)
	Z	=	0	mm (Q23)



Part program

```
%3600717 G71 *                                  ⎫
N10 D00 Q11 P01 +20 *                           ⎪
N20 D00 Q12 P01 +50 *                           ⎪ Begin the program; assign the coordinates for pre-
N30 D00 Q13 P01 +10 *                           ⎬ positioning the touch probe to Q parameters
N40 D00 Q21 P01 +50 *                           ⎪
N50 D00 Q22 P01 +10 *                           ⎭
N60 D00 Q23 P01 +0 *

N70 T0 G17 *
N80 G00 G40 G90 Z+100 M06 * .............................. Insert touch probe
N90 G55 P01 10 P02 Z- X+Q11 Y+Q12 Z+Q13 * ...... The Z coordinate probed in the negative direction is stored in
                                                 Q10 (1st point)
N100 X+Q21 Y+Q22 * ............................................... Move to auxiliary point for second pre-positioning
N110 G55 P01 20 P02 Z- X+Q21 Y+Q22 Z+Q23 * .... The Z coordinate probed in the negative direction is stored in
                                                 Q20 (2nd point)
N120 D02 Q01 P01 +Q20 P02 +Q10 ......................... Measure the height of the island and assign to Q1
N130 G38 * ............................................................... Q1 can be checked after the program run has been stopped
                                                 (see page 7-10)
N140 Z+100 M02 *
N9999 %3600717 G71 * ........................................... Retract the tool and end the program
```

7.8 Examples for Exercise

Rectangular pocket with corner rounding and tangential approach

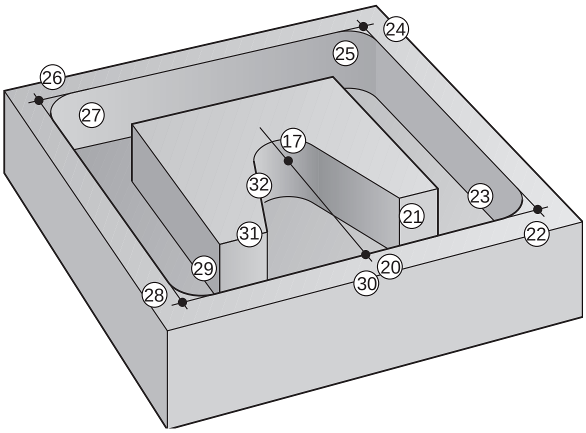
Pocket center coordinates

X = 50 mm (Q1)
Y = 50 mm (Q2)

Pocket length	X	=	90 mm	(Q3)
Pocket width	Y	=	70 mm	(Q4)
Working depth	Z	=	(–) 15 mm	(–Q5)
Corner radius	R	=	10 mm	(Q6)
Milling feed	F	=	200 mm/min	(Q7)

Note:

At corners 21 and 31 the workpiece will be machined slightly differently than shown in the drawing!



Part program

```
%360077 G71 *

N10 G30 G17 X+0 Y+0 Z–20 *              } Program start and workpiece blank
N20 G31 G90 X+100 Y+100 Z+0 *           }

N30 D00 Q01 P01 +50 *                   }
N40 D00 Q02 P01 +50 *                   }
N50 D00 Q03 P01 +90 *                   }
N60 D00 Q04 P01 +70 *                   } Assign the rectangular pocket data to Q parameters
N70 D00 Q05 P01 +15 *                   }
N80 D00 Q06 P01 +10 *                   }
N90 D00 Q07 P01 +200 *                  }

N100 G99 T1 L+0 R+5 *                   }
N110 T1 G17 S1000 *                     } Define and insert the tool
N120 G00 G40 G90 Z+100 M6 *             }

N130 D04 P01 Q13 P02 +Q03 P03 +2 *      } Enter half the pocket length and width for the paths of
N140 D04 P01 Q14 P02 +Q04 P03 +2 *      } traverse in blocks N200, N220, N300

N150 D04 P01 Q16 P02 +Q06 P03 +4 * ...................Rounding radius for smooth approach
N160 D04 P01 Q17 P02 +Q07 P03 +2 * ...................Feed rate in corners is half the rate for linear movement
N170 X+Q01 Y+Q02 M03 * .......................................Pre-position in X and Y (pocket center), spindle ON
N180 Z+2 * ...................................................................Pre-position over workpiece
N190 G01 Z–Q05 FQ07 * ..........................................Move to working depth Q5 (= –15 mm) with feed rate Q7
                                                                (= 100)

N200 G41 G91 X+Q13 G90 Y+Q02 *          } Approach the pocket in a tangential arc
N210 G26 RQ16 *                         }

N220 G91 Y+Q14 *                        }
N230 G25 RQ6 FQ17 *                     }
N240 X–Q3 *                             }
N250 G25 RQ6 FQ17 *                     }
N260 Y-Q4 *                             } Mill the frame of the rectangular pocket
N270 G25 RQ6 FQ17 *                     }
N280 X+Q3 *                             }
N290 G25 RQ6 FQ17 *                     }
N300 Y+Q14 *                            }

N310 G27 RQ16 *                         } Depart to pocket center in a tangential arc
N320 G00 G40 G90 X+Q1 Y+Q2 *            }

N330 Z+100 M02 * ....................................................Retract tool
N9999 %360077 G71 *
```

Bolt hole circles

Bore pattern 1 distributed over a full circle:

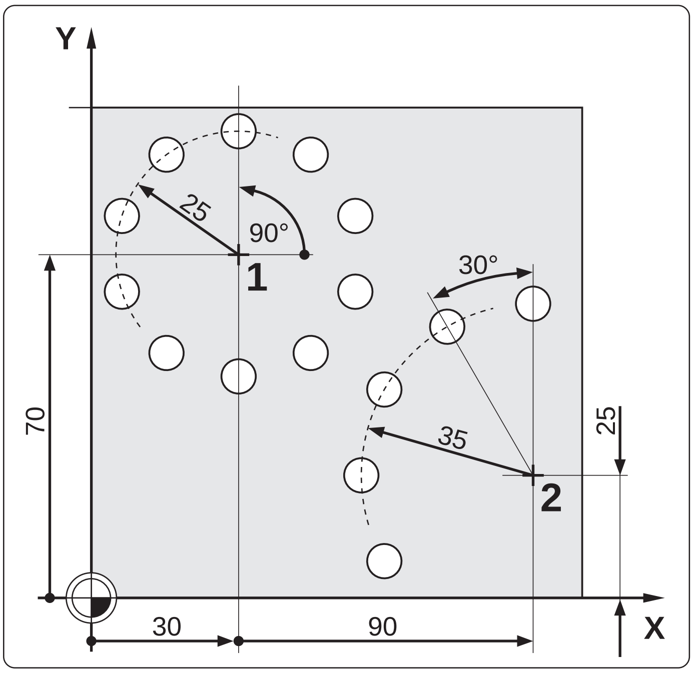
Entry values are listed below in program blocks N10 to N80.

Movements in the plane are programmed with polar coordinates.

Bore pattern 2 distributed over a circle sector:

Entry values are listed below in blocks N150 to N190; Q5, Q7 and Q8 remain the same.

The holes are executed with cycle G83: PECKING (see page 8-4)



Part program

```
%3600715 G71 * ....................................................... Load data for bolt hole circle 1
N10 D00 Q01 P01 +30 * ........................................... Circle center X coordinate
N20 D00 Q02 P01 +70 * ........................................... Circle center Y coordinate
N30 D00 Q03 P01 +11 * ........................................... Number of holes
N40 D00 Q04 P01 +25 * ........................................... Circle radius
N50 D00 Q05 P01 +90 * ........................................... Starting angle
N60 D00 Q06 P01 +0 * ............................................. Hole angle increment (0: distribute holes over 360°)
N70 D00 Q07 P01 +2 * ............................................. Setup clearance
N80 D00 Q08 P01 +15 * ........................................... Total hole depth
N90 G30 G17 X+0 Y+0 Z-20 *
N100 G31 G90 X+100 Y+100 Z+0 *
N110 G99 T1 L+0 R+4 *
N120 T1 G17 S2500 *
N130 G83 P01 -Q07 .................................................. Definition of the pecking cycle/setup clearance
          P02 -Q08......................................................... Total hole depth according to the load data
          P03 -5 ............................................................. Pecking depth
          P04 0 .............................................................. Dwell time
          P05 250 * ....................................................... Feed rate for pecking
N140 L1,0 * ................................................................ Call bolt hole circle 1
                                                                              Load data for bolt hole circle 2 (only re-enter changed data)
N150 D00 Q1 P01 +90 * ........................................... New circle center X coordinate
N160 D00 Q2 P01 +25 * ........................................... New circle center Y coordinate
N170 D00 Q3 P01 +5 * ............................................. New number of holes
N180 D00 Q4 P01 +35 * ........................................... New circle radius
N190 D00 Q6 P01 +30 * ........................................... New hole angle increment (not a full circle, 5 holes at 30°
                                                                              intervals)
N200 L1,0 * ................................................................ Call bolt hole circle 2
N210 G00 G40 G90 Z+200 M02 * ............................ End of main program
```

Continued...

```
N220 G98 L1 * ............................. Subprogram bolt hole circle
N230 D00 Q10 P01 +0 * ..................... Set the counter for finished holes
N240 D10 P01 +Q6 P02 +0 P03 10 * .......... If the hole angle increment has been entered, jump to LBL 10
N250 D04 Q6 P01 +360 P02 +Q3 * ............ Calculate the hole angle increment, distribute holes over 360°
N260 G98 L10 *
N270 D01 Q11 P01 +Q5 P02 +Q6 * ............ Calculate second hole position from the start angle and hole
                                            angle increment
N280 I+Q1 J+Q2 * .......................... Set pole at bolt hole circle center
N290 G10 G40 G90 R+Q4H+Q5 M03 * ........... Move in the plane to 1st hole
N300 G00 Z+Q7 M99 * ....................... Move in Z to setup clearance, call cycle
N310 D01 Q10 P01 +Q10 P02 +1 * ............ Count finished holes
N320 D09 P01 +Q10 P02 +Q3 P03 99 * ........ Finished?
N330 G98 L2 *
N340 G10 R+Q4 H+Q11 M99 * ................. Make a second and further holes
N350 D01 Q10 P01 +Q10 P02 +1 * ............ Count finished holes
N360 D01 Q11 P01 +Q11 P02 +Q6 * ........... Calculate angle for next hole (update)
N370 D12 P01 +Q10 P02 +Q3 P03 2 * ......... Not finished?
N380 G98 L99 *
N390 G00 G40 G90 Z+200 * .................. Retract in Z
N400 G98 L0 * ............................. End of subprogram, return jump to main program
N9999 %3600715 G71 *
```

Ellipse

X coordinate calculation: $X = a \cdot \cos \alpha$
Y coordinate calculation: $Y = b \cdot \sin \alpha$

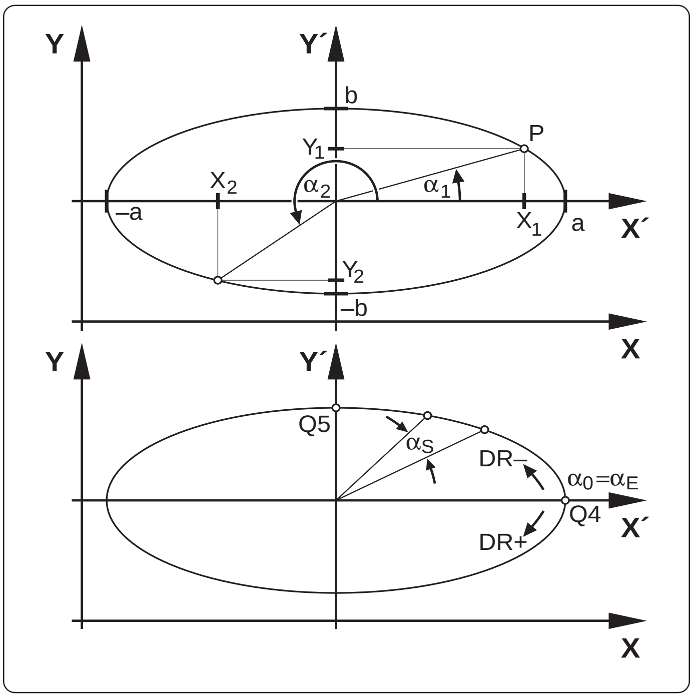
a, b : Semimajor and semiminor axes of the ellipse
α : Angle between the leading axis and the connecting line from P to the center of the ellipse

Process:

The points of the ellipse are calculated and connected by many short lines. The more points that are calculated and the shorter the lines between them, the smoother the curve.

The machining direction can be varied by changing the entries for start and end angles.

The input parameters are listed below in blocks N10 to N120 of the part program.



Part program

```
%376015 G71 * ........................................................ Load data
N10 D00 Q01 P01 +50 * .......................................... X coordinate for center of ellipse
N20 D00 Q02 P01 +50 * .......................................... Y coordinate for center of ellipse
N30 D00 Q03 P01 +50 * .......................................... Semiaxis in X
N40 D00 Q04 P01 +20 * .......................................... Semiaxis in Y
N50 D00 Q05 P01 +0 * ............................................ Start angle
N60 D00 Q06 P01 +360 * ........................................ End angle
N70 D00 Q07 P01 +40 * .......................................... Number of calculating steps
N80 D00 Q08 P01 +0 * ............................................ Rotational position
N90 D00 Q09 P01 +10 * .......................................... Depth
N100 D00 Q10 P01 +100 * ...................................... Plunging feed rate
N110 D00 Q11 P01 +350 * ...................................... Milling feed rate
N120 D00 Q12 P01 +2 * .......................................... Setup clearance Z
N130 G30 G17 X+0 Y+0 Z-20 * ................................ Definition of workpiece blank
N140 G31 G90 X+100 Y+100 Z+0 *
N150 G99 T1 L+0 R+2.5 *
N160 T1 G17 S2500 *
N170 G00 G40 G90 Z+100 * .................................... Retract in Z
N180 L10,0 * ............................................................. Call subprogram ellipse
N190 G00 Z+100 M02 * ........................................... Retract in Z, end of main program
```

Continued...

```
N200 G98 L10 *
N210 G54 X+Q1 Y+Q2 * ........................................ Shift datum to center of ellipse
N220 G73 G90 H+Q8 * ......................................... Activate rotation, if Q8 is loaded
N230 D02 Q35 P01 +Q6 P02 +Q5 *
N240 D04 Q35 P01 +Q35 P02 +Q7 * ............................. Calculate angle increment
N250 D00 Q36 P01 +Q5 * ...................................... Current angle for calculation = set start angle
N260 D00 Q37 P01 +0 * ....................................... Set counter for milled steps
N270 L11,0 * ................................................ Call subprogram for calculating the points of the ellipse
N280 G00 G40 X+Q21 Y+Q22 M03 * .............................. Move to start point in the plane
N290 Z+Q12 * ................................................ Rapid traverse in Z to setup clearance
N300 G01 Z-Q9 FQ10 * ........................................ Plunge to milling depth at plunging feed rate

N310 G98 L1 *
N320 D01 Q36 P01 +Q36 P02 +Q35 * ............................ Update the angle
N330 D01 Q37 P01 +Q37 P02 +1 * .............................. Update the counter
N340 L11,0 * ................................................ Call subprogram for calculating the points of the ellipse
N350 G01 X+Q21 Y+Q22 FQ11 * ................................. Move to next point
N360 D12 P01 +Q37 P02 +Q7 P03 1 * ........................... Not finished?

N370 G73 G90 H+0 * .......................................... Reset rotation
N380 G54 X+0 Y+0 * .......................................... Reset datum shift
N390 G00 G40 Z+Q12 * ........................................ Move in Z to setup clearance
N400 G98 L0 * ............................................... End of subprogram for milling the ellipse

N410 G98 L11 *
N420 D07 Q21 P01 +Q36 *
N430 D03 Q21 P01 +Q21 P02 +Q3 * ............................. Calculate X coordinate
N440 D06 Q22 P01 +Q36 *
N450 D03 Q22 P01 +Q22 P02 +Q4 * ............................. Calculate Y coordinate
N460 G98 L0 *
N9999 %376015 G71 *
```

Machining a hemisphere with an end mill

Notes on the program:

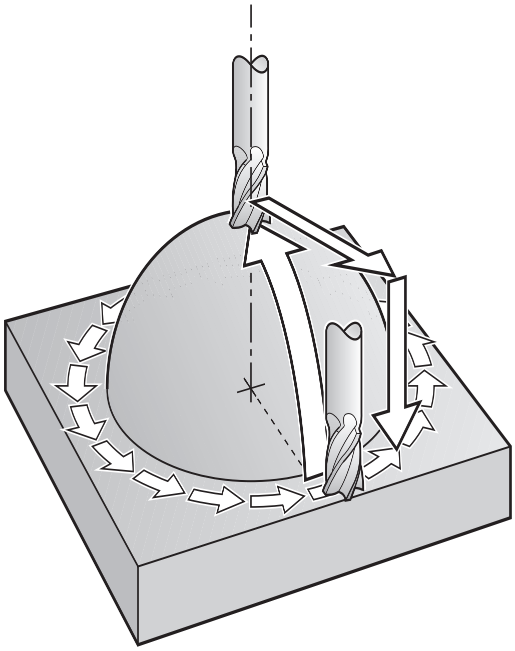
- The tool moves upwards in the ZX plane.
- You can enter an oversize in block N120 (Q12) if you want to machine the contour in several steps.
- The tool radius is automatically compensated with parameter Q108.

The program works with the following values:

• Solid angle:	Start angle	Q1
	End angle	Q2
	Increment	Q3
• Sphere radius		Q4
• Setup clearance		Q5
• Plane angle:	Start angle	Q6
	End angle	Q7
	Increment	Q8
• Center of sphere:	X coordinate	Q9
	Y coordinate	Q10
• Milling feed rate		Q11
• Oversize		Q12

The parameters additionally defined in the program have the following meanings:

- Q15: Setup clearance above the sphere
- Q21: Solid angle during machining
- Q24: Distance from center of sphere to center of tool
- Q26: Plane angle during machining
- Q108: TNC parameter with tool radius



Part program

```
%360712 G71 *
N10 D00 Q1 P01 + 90 *
N20 D00 Q2 P01 + 0 *
N30 D00 Q3 P01+ 5 *
N40 D00 Q4 P01 + 45 *
N50 D00 Q5 P01 + 2 *
N60 D00 Q6 P01+ 0 *                   Assign the sphere data to the param-
N70 D00 Q7 P01 + 360 *                eters
N80 D00 Q8 P01 + 5 *
N90 D00 Q9 P01 + 50 *
N100 D00 Q10 P01 + 50 *
N110 D00 Q11 P01 + 500 *
N120 D00 Q12 P01 + 0 *
N130 G30 G17 X+0 Y+0 Z–50 *
N140 G31 G90 X+100 Y+100 Z+0 *
N150 G99 T1 L+0 R+5 *                 Workpiece blank; define and insert tool
N160 T1 G17 S1000 *
N170 G00 G40 G90 Z+100 M06 *
N180 L 10,0 * ....................... Subprogram call
N190 G00 G40 G90 Z+100 M02 * ........ Retract tool; return jump to beginning of program
```

Continued...

```
N200 G98 L10 *
N200 D01 Q15 P01 +Q5 P02 +Q4 *
N220 D00 Q21 P01 +Q1 *                          Determine starting and calculation values
N230 D01 Q24 P01 +Q4 P02 +Q108 *
N240 D00 Q26 P01 +Q6 *

N250 G54 X+Q9 Y+Q10 Z-Q4 * ................................ Shift datum to center of sphere
N260 G73 G90 H+Q06 * ........................................... Rotation for program start (starting plane angle)

N270 I+0 J+0 * .......................................................... Pole for pre-positioning
N280 G10 G40 G90 R+Q24 H+Q6 * .......................... Pre-positioning before machining

N290 G98 L1 *
N300 K+0 I+Q108 *
N310 G01 Y+0 Z+0 FQ11 * ....................................... Pre-positioning at the beginning of each arc
N320 G98 L2 *

N330 G11 G40 R+Q4 H+Q21 FQ11 *
N340 D02 Q21 P01 +Q21 P02 +Q03 *                Mill the sphere upward until the highest point is reached
N350 D11 P01 +Q21 P02 +Q02 P03 2 *

N360 R+Q04 H+Q02 *
N370 G01 Z+Q15 F1000 *                          Mill the highest point and then retract the tool
N380 G00 G40 X+Q24 *

N390 D01 Q26 P01 +Q26 P02 +Q08 * ...................... Prepare the next rotation increment
N400 D00 Q21 P01 Q01 * ......................................... Reset solid angle for machining to the starting value

N410 G73 G90 H+Q26 *                            Rotate the coordinate system about the Z axis until plane end
N420 D12 P01 +Q26 P02 +Q07 P03 1 *              angle is reached
N430 D09 P01 +Q26 P02 + Q07 P03 1 *

N440 G73 G90 H+0 * ................................................. Reset rotation
N450 G54 X+0 Y+0 Z+0 * ......................................... Reset datum shift

N460 G98 L0 * .......................................................... End of subprogram
N9999 %360712 G71 *
```

8.1 General Overview of Cycles

Frequently recurring machining sequences comprising several steps are stored in the TNC memory as cycles. Coordinate transformations and other special functions are also available as cycles.

The cycles are divided into several groups:

- **Simple fixed cycles** such as pecking and tapping as well as the milling operations slot milling, circular pocket milling and rectangular pocket milling.
- **SL (Subcontour List) cycles**, which allow machining of relatively complex contours composed of several overlapping subcontours.
- **Coordinate transformation cycles** which enable datum shift, rotation, mirror image, enlarging and reducing for various contours.
- **Special cycles** such as dwell time, program call and oriented spindle stop.

Programming a cycle

Defining a cycle

Select the desired cycle and program it in the dialog by entering the appropriate G function. The following example shows how to define any cycle:

G 8 5 ENT	Select a cycle, for example RIGID TAPPING.
SETUP CLEARANCE?	
e.g. +/– 2 ENT	Enter setup clearance, for example –2 mm.
TOTAL HOLE DEPTH?	
e.g. +/– 3 0 ENT	Enter total hole depth, for example –30 mm.
THREAD PITCH?	
e.g. 0 . 7 5 END □	Enter thread pitch, for example 0.75 mm.

*Resulting NC block: G85 P01–2 P02–30 P03+0.75 **

Cycle call

The following cycles become effective immediately upon being defined in the part program:

- Coordinate transformation cycles
- Dwell time
- The SL cycle G37 CONTOUR GEOMETRY

All other cycles must be called separately. Further information on cycle calls is provided in the descriptions of the individual cycles.

If the cycle is to be executed after the block in which it was called up, program the cycle call

- with G79
- with the miscellaneous function M99.

If the cycle is to be run after every positioning block, it must be called with the miscellaneous function M89 (depending on the machine parameters).

M89 is cancelled with M99.

Prerequisites:

The following data must be programmed before a cycle call:

- Blank form for graphic display
- Tool call
- Positioning block for starting position X, Y
- Positioning block for starting position Z (setup clearance)
- Direction of rotation of the spindle (miscellaneous functions M3/M4)
- Cycle definition.

Dimensions in the tool axis

The dimensions for tool axis movement are always referenced to the position of the tool at the time of the cycle call and interpreted by the control as incremental dimensions. It is not necessary to program G91.

The algebraic signs for SETUP CLEARANCE, TOTAL HOLE DEPTH and JOG INCREMENT define the working direction. They must be entered identically (usually negative).

The TNC assumes that at the beginning of the cycle the tool is positioned over the workpiece at the clearance height.

8.2 Simple Fixed Cycles

PECKING G83

Process:

- The tool drills at the entered feed rate to the first pecking depth.
- The tool is then retracted at rapid traverse to the starting position and advances again to the first pecking depth, minus the advanced stop distance t (see calculations).
- The tool advances with another infeed at the programmed feed rate.
- These steps are repeated until the programmed total hole depth is reached.
- After a dwell time at the bottom of the hole, the tool is retracted to the starting position at rapid traverse for chip breaking.

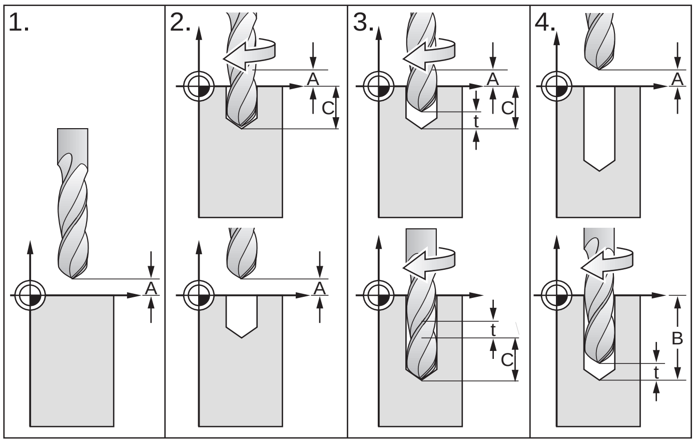


Fig. 8.1: PECKING cycle

Input data

- SETUP CLEARANCE Ⓐ:
  Distance between tool tip (at starting position) and workpiece surface.
- TOTAL HOLE DEPTH Ⓑ:
  Distance between workpiece surface and bottom of hole (tip of drill taper).
- PECKING DEPTH Ⓒ:
  Infeed per cut.
  If the TOTAL HOLE DEPTH equals the PECKING DEPTH, the tool will drill to the programmed hole depth in one operation. The PECKING DEPTH does not have to be a multiple of the TOTAL HOLE DEPTH. If the PECKING DEPTH is greater than the TOTAL HOLE DEPTH, the tool only advances to the TOTAL HOLE DEPTH.
- DWELL TIME:
  Length of time the tool remains at the total hole depth for chip breaking.

Calculations

The advanced stop distance is automatically calculated by the control:

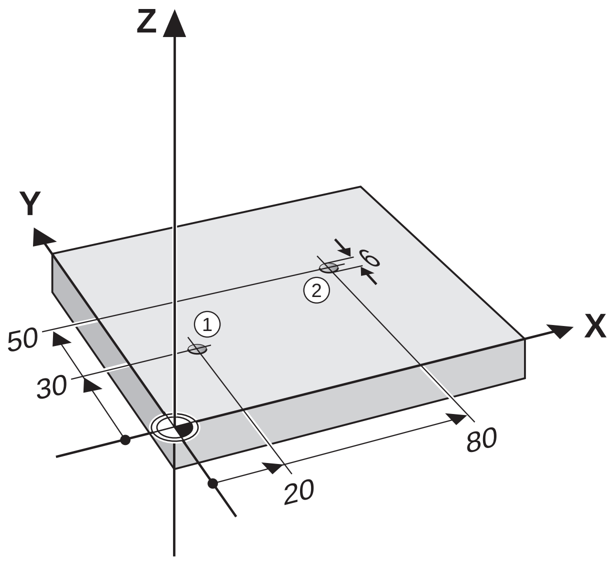
- Total hole depth up to 30 mm: t = 0.6 mm
- Total hole depth over 30 mm: t = Total hole depth / 50
  maximum advanced stop distance: 7 mm

Example: Pecking

Hole coordinates:

①	X	=	20 mm	Y	=	30 mm
②	X	=	80 mm	Y	=	50 mm

Hole diameter:	6	mm
Setup clearance:	2	mm
Total hole depth:	15	mm
Pecking depth:	10	mm
Dwell time:	1	s
Feed rate:	80	mm/min



PECKING cycle in a part program

```
%S85I G71 * .................................................................... Begin program
N10 G30 G17 X+0 Y+0 Z–20 * ........................................ Define workpiece blank
N20 G31 G90 X+100 Y+100 Z+0 *
N30 G99 T1 L+0 R+3 * .................................................... Tool definition
N40 T1 G17 S1200 * ....................................................... Tool call
N50 G83 P01 –2 P02 –15 P03 –10 P04 1 P05 80 * .......... Cycle definition PECKING
N60 G00 G40 G90 Z+100 M06 * .................................... Retract the spindle, insert the tool
N70 X+20 Y+30 M03 * .................................................... Pre-positioning for first hole, spindle ON
N80 Z+2 M99 * ............................................................... Pre-positioning in Z to setup clearance, cycle call
N90 X+80 Y+50 M99 * .................................................... Move to second hole, cycle call
N100 Z+100 M02 * ......................................................... Retract tool and end program
N9999 %S85I G71 *
```

TAPPING with floating tap holder G84

Process

- The thread is cut in one pass.
- When the tool reaches the total hole depth, the direction of spindle rotation is reversed. After the programmed dwell time the tool is retracted to the starting position.
- At the starting position, the direction of rotation is reversed once again.

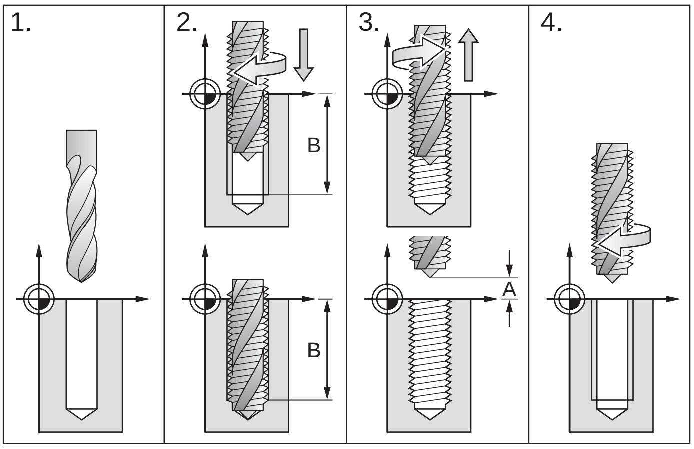


Fig. 8.2: TAPPING cycle

Required tool

A floating tap holder is required for tapping. The floating tap holder compensates the tolerances for feed rate and spindle speed during the tapping process.

Input data

- SETUP CLEARANCE Ⓐ:
  Distance between tool tip (starting position) and workpiece surface.
  Standard value: 4x thread pitch.
- TOTAL HOLE DEPTH Ⓑ (thread length):
  Distance between workpiece surface and end of thread.
- DWELL TIME:
  Enter a dwell time between 0 and 0.5 seconds to prevent wedging of the tool when retracted. (Further information is available from the machine tool builder.)
- FEED RATE F:
  Traversing speed of the tool during tapping.

Calculations

The feed rate is calculated as follows:

$F = S \times p$

F: Feed rate (mm/min)
S: Spindle speed (rpm)
p: Thread pitch (mm)

- When a cycle is being run, the spindle speed override control is disabled. The feed rate override control is only active within a limited range (preset by the machine tool builder).
- For tapping right-hand threads activate the spindle with M3; for left-hand threads use M4.

Example: Tapping with a floating tap holder

Cutting an M6 thread at 100 rpm

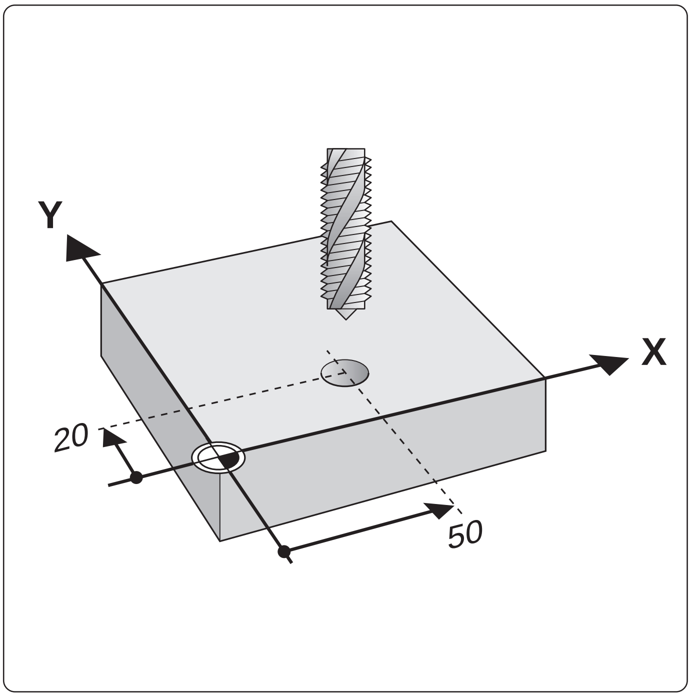
Coordinates of the hole:

X = 50 mm Y = 20 mm

Pitch p = 1 mm

F = S x p ➞ F = 100 · 1 = 100 mm/min

Setup clearance:	3	mm
Thread depth:	20	mm
Dwell time:	0.4	s
Feed rate:	100	mm/min



TAPPING cycle in a part program

```
%S87I G71 * ............................................................. Begin program
N10 G30 G17 X+0 Y+0 Z–20 * .................................. Define workpiece blank
N20 G31 G90 X+100 Y+100 Z+0 *
N30 G99 T1 L+0 R+3 * ............................................... Tool definition
N40 T1 G17 S100 * .................................................... Tool call
N50 G84 P01 –5 P02 –20 P03 0.4 P04 100 * ............. Cycle definition TAPPING
N60 G00 G40 G90 Z+100 M06 * ............................... Retract the spindle, insert the tool
N70 X+50 Y+20 M03 * .............................................. Pre-positioning in the X/Y plane, spindle ON
N80 Z+3 M99 * .......................................................... Pre-positioning in Z to setup clearance, cycle call
N90 Z+100 M02 * ...................................................... Retract tool and end program
N9999 %S87I G71 *
```

RIGID TAPPING G85

Process

The thread is cut without a floating tap holder in one or several passes.

Advantages over tapping with a floating tap holder:

- Higher machining speeds
- Repeated tapping of the same thread; repetitions are made possible by spindle orientation to the 0° position during cycle call (depending on machine parameters)
- Increased traverse range of the spindle axis

> Machine and control must be specially prepared by the machine manufacturer to enable rigid tapping.

Input data

- SETUP CLEARANCE Ⓐ:
  Distance between tool tip (starting position) and workpiece surface.
- TAPPING DEPTH Ⓑ:
  Distance between workpiece surface (beginning of thread) and end of thread
- THREAD PITCH Ⓒ:
  The sign differentiates between right-hand and left-hand threads:
  + = Right-hand thread
  – = Left-hand thread

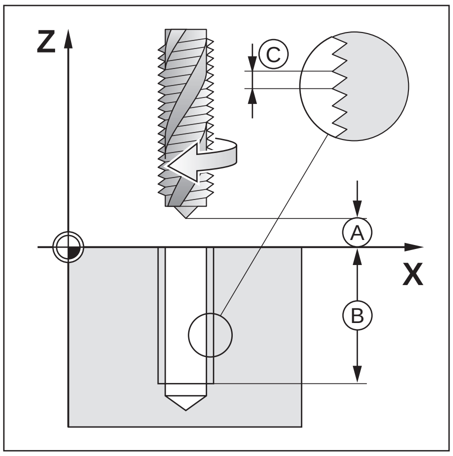


Fig. 8.3: Input data for the RIGID TAPPING cycle

> The control calculates the feed rate from the spindle speed. If the spindle speed override knob is turned during tapping, the control automatically adjusts the feed rate accordingly. The feed rate override is disabled.

SLOT MILLING G74

Process

Roughing process:

- The tool penetrates the workpiece from the starting position and mills in the longitudinal direction of the slot.
- After downfeed at the end of the slot, milling is performed in the opposite direction.
  These steps are repeated until the programmed milling depth is reached.

Finishing process:

- The control advances the tool in a quarter circle at the bottom of the slot by the remaining finishing cut. The tool subsequently climb mills the contour (with M3).
- At the end of the cycle, the tool is retracted in rapid traverse to the setup clearance.
  If the number of infeeds was odd, the tool returns to the starting position at the level of the setup clearance.

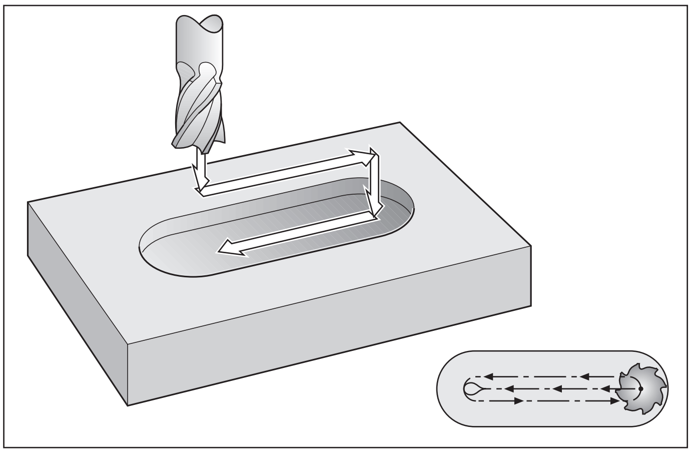


Fig. 8.4: SLOT MILLING cycle

Required tool

This cycle requires a center-cut end mill (ISO 1641). The cutter diameter must not be larger than the width of the slot and not smaller than half the width of the slot. The slot must be parallel to an axis of the current coordinate system.

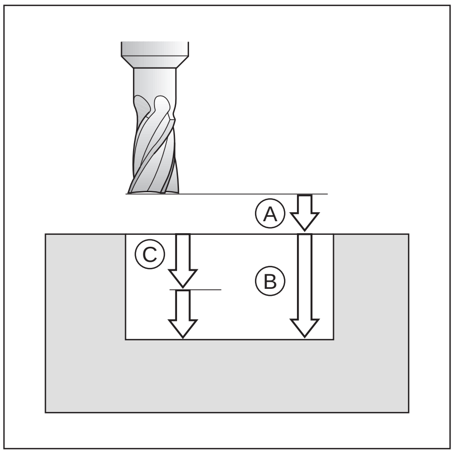


Fig. 8.5: Infeeds and distances for the SLOT MILLING cycle

Input data

- SETUP CLEARANCE Ⓐ
- MILLING DEPTH Ⓑ: Depth of the slot
- PECKING DEPTH Ⓒ
- FEED RATE FOR PECKING:
  Traversing speed of the tool during penetration.
- FIRST SIDE LENGTH Ⓓ:
  Length of the slot. Specify the sign to determine the first milling direction.
- SECOND SIDE LENGTH Ⓔ:
  Width of the slot
- FEED RATE:
  Traversing speed of the tool in the working plane.

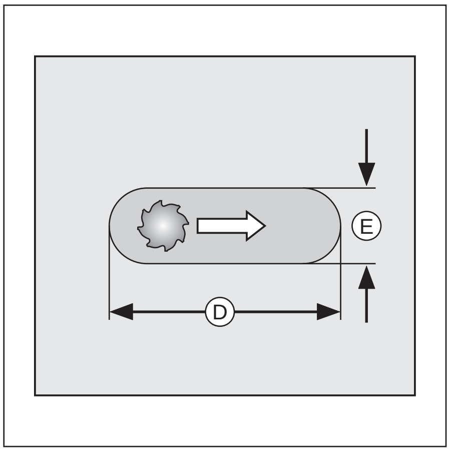


Fig. 8.6: Side lengths of the slot

Example: Slot milling

A horizontal slot 50 mm x 10 mm and a vertical slot 80 mm x 10 mm are to be milled.

The starting position takes into account the tool radius in the longitudinal direction of the slot.

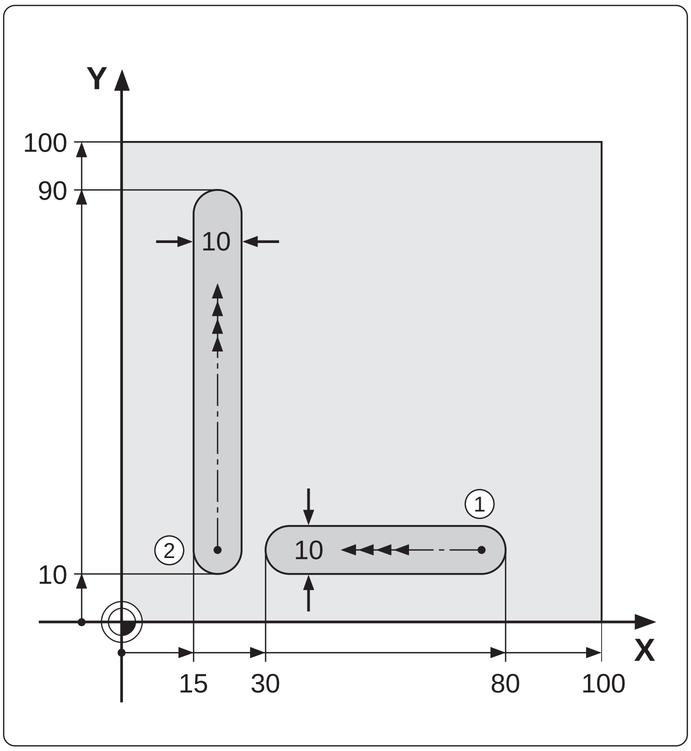
Starting position slot ①:
X = 76 mm   Y = 15 mm

Starting position slot ②:
X = 20 mm   Y = 14 mm

SLOT DEPTHS:	15 mm
Setup clearances:	2 mm
Milling depths:	15 mm
Pecking depths:	5 mm
Feed rate for pecking:	80 mm/min

	①	②
Slot length 1st milling direction	50 mm –	80 mm +

Slot widths:	10 mm
Feed rate:	120 mm/min



SLOT MILLING cycle in a part program

```
%S810I G71 * ........................................................ Begin program
N10 G30 G17 X+0 Y+0 Z–20 * ................................. Define workpiece blank
N20 G31 G90 X+100 Y+100 Z+0 *
N30 G99 T1 L+0 R+4 * ............................................ Tool definition
N40 T1 G17 S2000 * ................................................ Tool call
N50 G74 P01 –2 P02 –15 P03 –5 P04 80 P05 X–50
P06 Y+10 P07 120 * ................................................. Define slot parallel to X axis
N60 G00 G40 G90 Z+100 M06 * .............................. Retract the spindle, insert the tool
N70 X+76 Y+15 M03 * ............................................ Move to starting position, spindle ON
N80 Z+2 M99 * ........................................................ Pre-positioning in Z to setup clearance, cycle call  ①
N90 G74 P01 –2 P02 –15 P03 –5 P04 80 P05 Y+80
P06 X+10 P07 120 * ................................................. Define slot parallel to Y axis
N100 X+20 Y+14 M99 * .......................................... Move to starting position, cycle call ②
N110 Z+100 M02 * .................................................. Retract tool and end program
N9999 %S810I G71 *
```

POCKET MILLING G75/G76

Process

The rectangular pocket milling cycle is a roughing cycle, in which

- the tool penetrates the workpiece at the starting position (pocket center)
- the tool subsequently follows the programmed path at the specified feed rate (see Fig. 8.9).

The cutter begins milling in the positive axis direction of the longer side. With square pockets, the cutter begins in the positive Y direction. At the end of the cycle, the tool returns to the starting position.

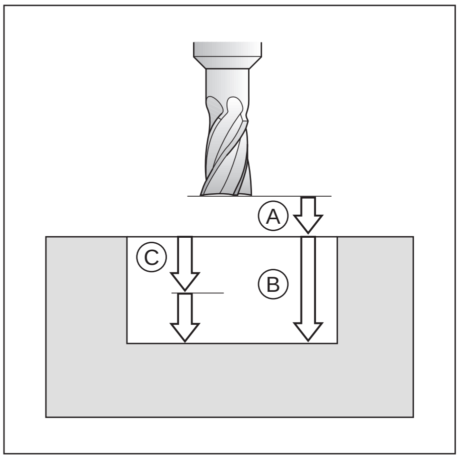


Fig. 8.7: Infeeds and distances for the POCKET MILLING cycle

Requirements / Limitations

This cycle requires a center-cut end mill (ISO 1641) or a separate pilot drilling operation at the pocket center. The pocket sides are parallel to the axes of the coordinate system.

Direction of rotation for roughing out

Clockwise direction of rotation: G75
Counterclockwise direction of rotation: G76

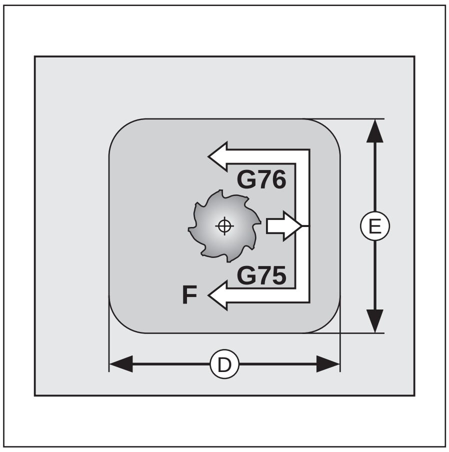


Fig. 8.8: Side lengths of the pocket

Input data

- Setup clearance Ⓐ
- Milling depth Ⓑ
- Pecking depth Ⓒ
- FEED RATE FOR PECKING:
  Traversing speed of the tool during penetration.
- FIRST SIDE LENGTH Ⓓ:
  Length of the pocket, parallel to the first main axis of the working plane.
- SECOND SIDE LENGTH Ⓔ:
  Width of the pocket
  The signs of the side lengths are always positive.
- FEED RATE:
  Traversing speed of the tool in the working plane.

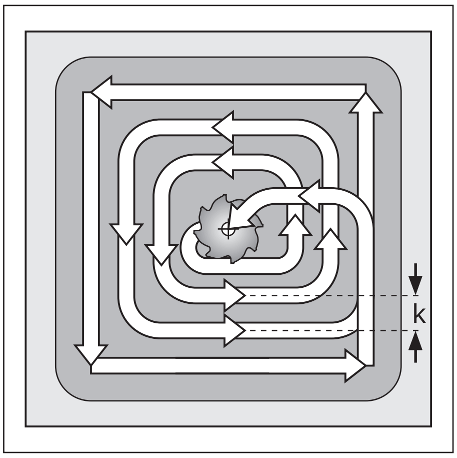


Fig. 8.9: Tool path for roughing out

Calculations

Stepover factor k:

k = K x R

K: Overlap factor (preset by the machine tool builder)
R: Cutter radius

Rounding radius

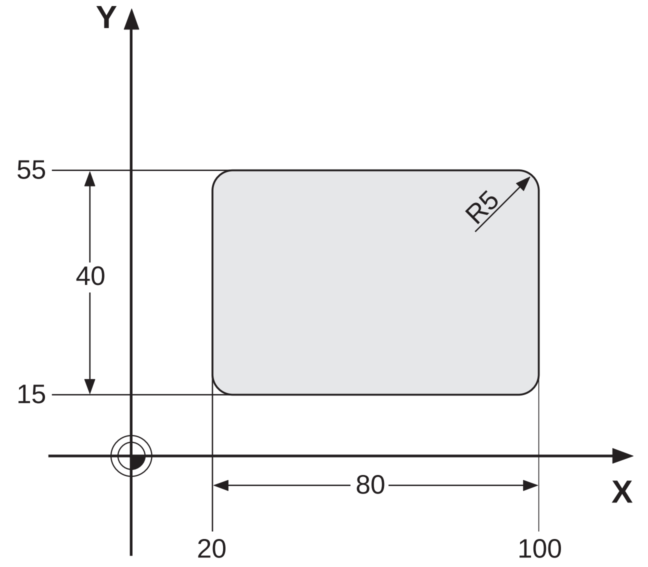
The pocket corners are rounded with the cutter radius.

Example: Rectangular pocket milling

Pocket center coordinates:

X = 60 mm Y = 35 mm

Setup clearance:	2	mm
Milling depth:	10	mm
Pecking depth:	4	mm
Feed rate for pecking:	80	mm/min
First side length:	80	mm
Second side length:	40	mm
Milling feed rate:	100	mm/min
Direction of the cutter path:	+	



POCKET MILLING cycle in a part program

```
%S812I G71 * ............................................................ Begin program
N10 G30 G17 X+0 Y+0 Z–20 * .................................. Define workpiece blank
N20 G31 G90 X+110 Y+100 Z+0 *
N30 G99 T1 L+0 R+5 * ............................................. Tool definition
N40 T1 G17 S2000 * ................................................. Tool call
N50 G76 P01 –2 P02 –10 P03 –4 P04 80 P05 X+80
P06 Y+40 P07 100 * .................................................. Cycle definition POCKET MILLING
N60 G00 G40 G90 Z+100 M06 * ............................... Retract the spindle, insert the tool
N70 X+60 Y+35 M03 * ............................................. Move to starting position (pocket center), spindle ON
N80 Z+2 M99 * .......................................................... Pre-positioning in Z to setup clearance, cycle call
N90 Z+100 M02 * ...................................................... Retract tool and end program
N9999 %S812I G71 *
```

CIRCULAR POCKET MILLING G77/G78

Process

- Circular pocket milling is a roughing cycle. The tool penetrates the workpiece from the starting position (pocket center).
- The cutter then follows a spiral path at the programmed feed rate (see Fig. 8.10). The stepover factor is determined by the value of k (see POCKET MILLING cycle G75/G76: calculations).
- The process is repeated until the programmed milling depth is reached.
- At the end of the cycle the tool returns to the starting position.

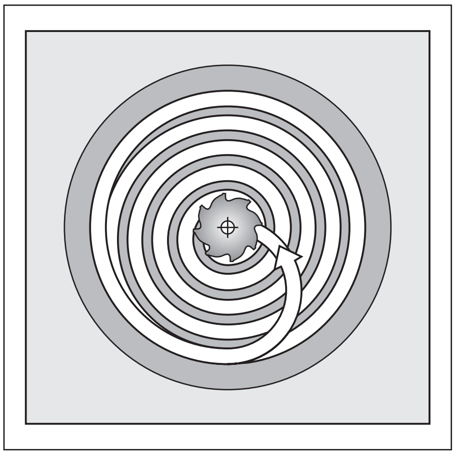


Fig. 8.10: Tool path for roughing out

Required tool

This cycle requires a center-cut end mill (ISO 1641) or a separate pilot drilling operation at the pocket center.

Direction of rotation for roughing out

Clockwise direction of rotation G77
Counterclockwise direction of rotation G78

Input data

- SETUP CLEARANCE Ⓐ
- MILLING DEPTH Ⓑ: DEPTH of the pocket
- PECKING DEPTH Ⓒ
- FEED RATE FOR PECKING:
  Traversing speed of the tool during penetration.
- CIRCLE RADIUS Ⓡ:
  Radius of the circular pocket.
- FEED RATE:
  Traversing speed of the tool in the working plane.

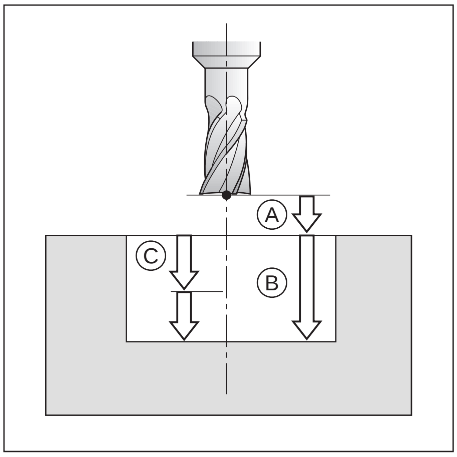


Fig. 8.11: Distances and infeeds with CIRCULAR POCKET MILLING

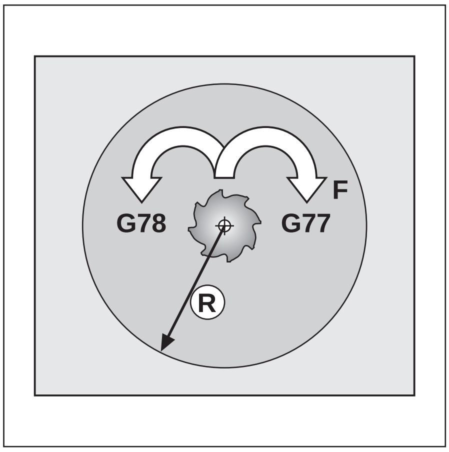


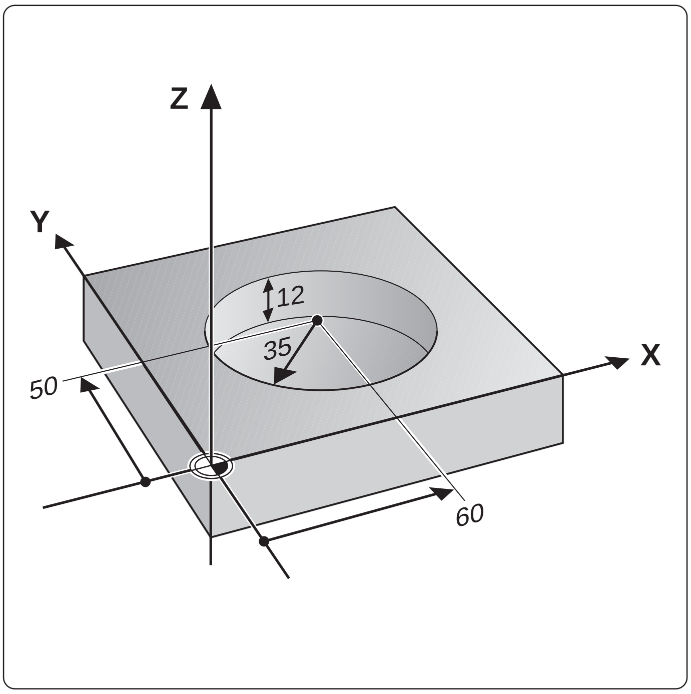
Fig. 8.12: Direction of the cutter path

Example: Milling a circular pocket

Pocket center coordinates:

X = 60 mm Y = 50 mm

Setup clearance:	2	mm
Milling depth:	12	mm
Pecking depth:	6	mm
Feed rate for pecking:	80	mm/min
Circle radius:	35	mm
Milling feed rate:	100	mm/min
Direction of the cutter path:	–	



CIRCULAR POCKET MILLING cycle in a part program

```
%S814I G71 * ........................................................... Begin program
N10 G30 G17 X+0 Y+0 Z–20 * .................................. Define workpiece blank
N20 G31 G90 X+100 Y+100 Z+0 *
N30 G99 T1 L+0 R+4 * ............................................ Tool definition
N40 T1 G17 S2000 * ................................................ Tool call
N50 G77 P01 –2 P02 –12 P03 –6 P04 80 P05 35
P06 100 * ................................................................. Cycle definition CIRCULAR POCKET MILLING
N60 G00 G40 G90 Z+100 M06 * .............................. Retract the spindle, insert the tool
N70 X+60 Y+50 M03 * ............................................ Move to starting position (pocket center), spindle ON
N80 Z+2 M99 * ........................................................ Pre-positioning in Z to setup clearance, cycle call

N90 Z+100 M02 * ..................................................... Retract tool and end program
N9999 %S814I G71 *
```

8.3 SL Cycles

Subcontour list (SL) cycles are very powerful cycles that enable you to mill any required contour. They are characterized by the following features:

- A contour can consist of superimposed subcontours. Pockets and islands compose the subcontours.
- The subcontours are programmed as subprograms.
- The control automatically superimposes the subcontours and calculates the points of intersection of the subcontours with each other.

Cycle G37: CONTOUR GEOMETRY contains the subcontour list and is a purely geometric cycle, containing no cutting data or infeed values.

The machining data are defined in the following cycles:

- PILOT DRILLING G56
- ROUGH-OUT G57
- CONTOUR MILLING G58/G59

Each subprogram defines whether radius compensation G41 or G42 applies. The sequence of points determines the direction of rotation in which the contour is to be machined. The control deduces from these data whether the specific subprogram describes a pocket or an island:

- For a pocket the tool path is inside the contour
- For an island the tool path is outside the contour

- The way the SL contour is machined is determined by MP 7420.
- We recommend a graphical test run before you machine the part. This will show whether all contours were correctly defined.
- All coordinate transformations are allowed in the subprograms for the subcontours.
- F and M words are ignored in the subprograms for the subcontours.

The following examples will at first use only the ROUGH-OUT cycle. Later, as the examples become more complex, the full range of possibilities of this group of cycles will be illustrated.

Programming parallel axes

Machining operations can also be programmed in parallel axes as SL cycles. The parallel axes must lie in the working plane. (In this case, graphic simulation is not available).

Input

Parallel axes must be programmed in the first coordinate block (positioning block, I,J,K block) of the first subprogram that is called with cycle G37: CONTOUR GEOMETRY. All other coordinates are then ignored.

CONTOUR GEOMETRY G37

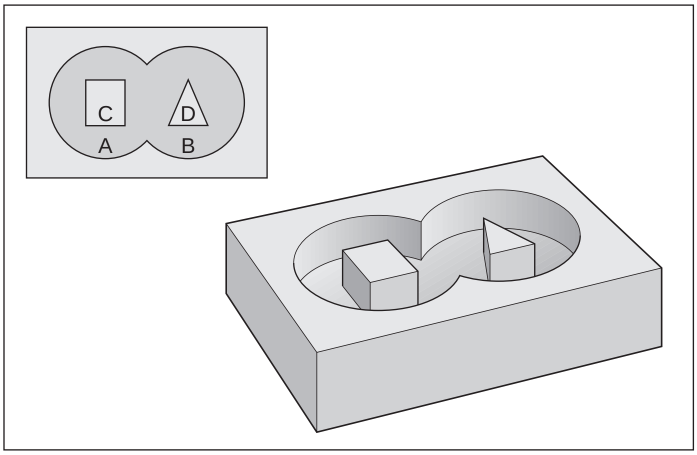


Fig. 8.13: Example of an SL contour: A, B = pockets; C, D = islands

Application

Cycle G37: CONTOUR GEOMETRY contains the list of subcontours that make up the complete contour.

Input data

Enter the LABEL numbers of the subprograms. A maximum of 12 subprograms can be listed.

Effect

Cycle G37 becomes effective as soon as it is defined.

Example:

```
G99 T3 L+0 R+3.5 *
T3 G17 S1500 * ......................................................... Working plane perpendicular to Z axis
G37 P01 1 P02 2 P03 3 *

  .
  .
G00 G40 Z+100 M2 *

  .
  .
G98 L1 ....................................................................... First contour label of the CONTOUR GEOMETRY cycle G37
G01 G42 X+0 Y+10 .................................................. Machining in the X/Y plane
X+20 Y+10
I+50 J+50

  .
  .
  .
```

ROUGH-OUT G57

Process

Cycle G57 specifies the cutting path and partitioning.

- The tool is positioned in the tool axis above the first infeed point, taking the finishing allowance into account.
- Then the tool penetrates into the workpiece at the programmed feed rate for pecking.

Milling the contour:

- The tool mills the first subcontour at the specified feed rate, taking the finishing allowance into account.
- When the tool returns to the infeed point, it is advanced to the next pecking depth.

This process is repeated until the programmed milling depth is reached.

- Further subcontours are milled in the same manner.

Roughing out pockets:

- After milling the contour the pocket is roughed out. The stepover is defined by the tool radius. Islands are jumped over.
- If necessary, pockets can be cleared out with several downfeeds.
- At the end of the cycle the tool returns to the setup clearance.

Required tool

This cycle requires a center-cut end mill (ISO 1641) if the pocket is not separately pilot drilled or if the tool must repeatedly jump over contours.

Input data

- SETUP CLEARANCE Ⓐ
- MILLING DEPTH Ⓑ
- PECKING DEPTH Ⓒ
- FEED RATE FOR PECKING:
  Traversing speed of the tool during penetration.
- FINISHING ALLOWANCE Ⓓ:
  Allowance in the working plane (positive number).
- ROUGH-OUT ANGLE (α):
  Feed direction for roughing out. The rough-out angle is relative to the angle reference axis and can be set such that the resulting cuts are as long as possible with few cutting movements.
- FEED RATE:
  Traversing speed of the tool in the working plane.

Machine parameters determine whether

- the contour is first milled and then surface machined, or vice-versa
- the contour is milled conventionally or by climb milling
- all pockets are first roughed out to the full milling depths and then contour milled, or vice-versa
- contour milling and roughing out are performed together for each pecking depth.

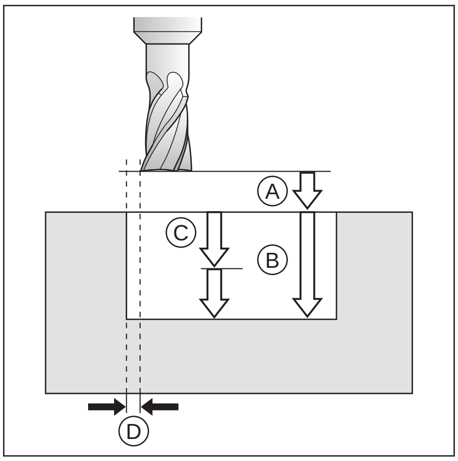


Fig. 8.14: Infeeds and distances with the ROUGH-OUT cycle

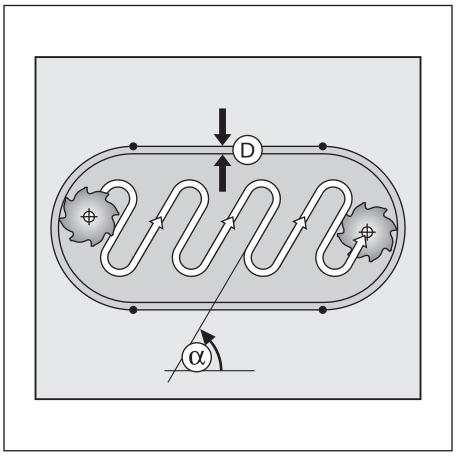


Fig. 8.15: Tool path for rough-out

Example: Roughing out a rectangular island

Rectangular island with rounded corners

Tool: center-cut end mill (ISO 1641), radius 5 mm.

Coordinates of the island corners:

	X	Y
①	70 mm	60 mm
②	15 mm	60 mm
③	15 mm	20 mm
④	70 mm	20 mm

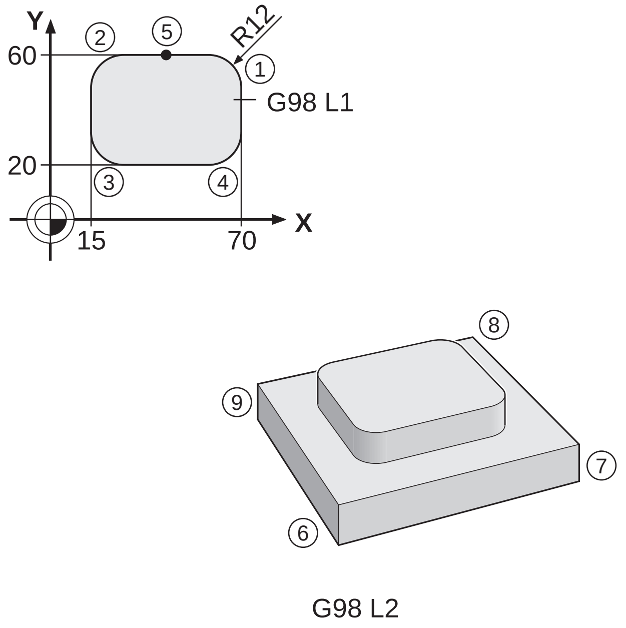
Coordinates of the auxiliary pocket:

	X	Y
⑥	–5 mm	–5 mm
⑦	105 mm	–5 mm
⑧	105 mm	105 mm
⑨	–5 mm	105 mm

Starting point for machining:

⑤ X = 40 mm Y = 60 mm

Setup clearance:	2	mm
Milling depth:	15	mm
Pecking depth:	8	mm
Feed rate for pecking:	100	mm/min
Finishing allowance:	0	
Rough-out angle:	0^0	
Feed rate for milling:	500	mm/min



Cycle in a part program

```
%S818I G71 * ........................................... Begin program
N10 G30 G17 X+0 Y+0 Z–20 * ............................. Define workpiece blank
N20 G31 X+100 Y+100 Z+0 *
N30 G99 T1 L+0 R+3 * ................................... Tool definition
N40 T1 G17 S2500 * ..................................... Tool call
N50 G37 P01 2 P02 1 * .................................. Define in cycle CONTOUR GEOMETRY that the contour
                                                         elements are described in subprograms 1 and 2
N60 G57 P01 –2 P02 –15 P03 –8 P04 100 P05 +0
P06 +0 P07 500 * ....................................... Cycle definition ROUGH-OUT
N70 G00 G40 G90 Z+100 M06 * ............................ Retract the spindle, insert the tool
N80 X+40 Y+50 M03 * .................................... Pre-positioning in X and Y, spindle ON
N90 Z+2 M99 * .......................................... Pre-positioning in Z to setup clearance, cycle call
N100 Z+100 M02 *

N110 G98 L1 *                                            Subprogram 1:
N120 G01 G42 X+40 Y+60 *                                 Geometry of the island
N130 X+15 *                                              (From radius compensation G42 and counterclockwise
                                                         machining, the control concludes that the contour element is
N150 Y+20 *                                            an island)
N160 G25 R12 *
N170 X+70 *
N180 G25 R12 *
N190 Y+60 *
N200 G25 R12 *
N210 X+40 *
N220 G98 L0 *

N230 G98 L2 *                                            Subprogram 2:
N240 G01 G41 X-5 Y-5 *                                   Geometry of the auxiliary pocket:
N250 X+105 *                                             External limitation of the machining surface
N260 Y+105 *                                             (From radius compensation G41 and counterclockwise
N270 X–5  *                                              machining, the control concludes that the contour element is
N280 Y–5 *                                               a pocket)
N290 G98 L0 *
N9999 %S818I G71 *
```

Overlapping contours

Pockets and islands can be overlapped to form a new contour. The area of a pocket can thus be enlarged by another pocket or reduced by an island.

Starting position

Machining begins at the starting position of the first pocket in cycle G37 CONTOUR GEOMETRY. The starting position should be located as far as possible from the overlapping contours.

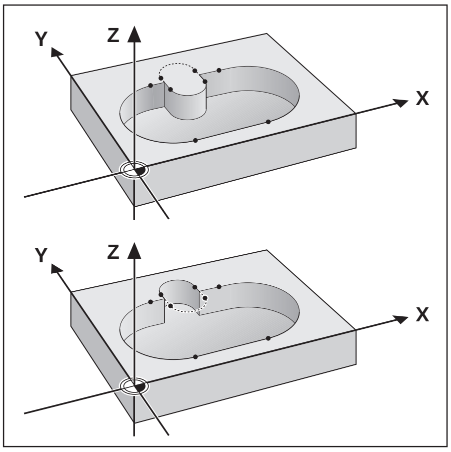


Fig. 8.16: Examples of overlapping contours

Example: Overlapping pockets

Machining begins with the first contour label defined in block 6. The first pocket must begin outside the second pocket.

Inside machining with a center-cut end mill (ISO 1641), tool radius 3 mm.

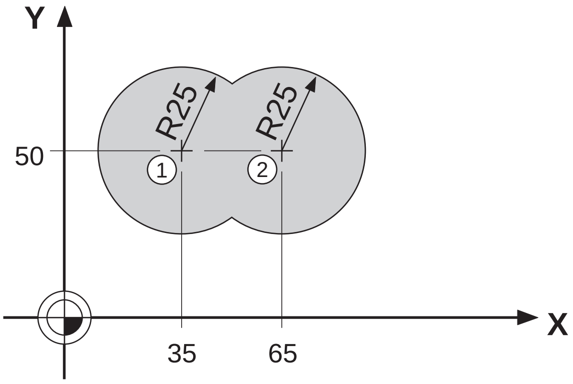
Coordinates of the circle centers:

①	X = 35 mm	Y = 50 mm
②	X = 65 mm	Y = 50 mm

Circle radii

R = 25 mm

Setup clearance:	2 mm
Milling depth:	10 mm
Pecking depth:	5 mm
Feed rate for pecking:	500 mm/min
Finishing allowance:	0
Rough-out angle:	0
Milling feed rate:	500 mm/min



Continued...

Cycle in a part program

```
%S820I G71 * ........................................................... Begin program
N10 G30 G17 X+0 Y+0 Z–20 * .................................. Define workpiece blank
N20 G31 X+100 Y+100 Z+0 *
N30 G99 T1 L+0 R+3 * ............................................. Tool definition
N40 T1 G17 S2500 * ................................................. Tool call
N50 G37 P01 1 P02 2 * ............................................ Define in cycle CONTOUR GEOMETRY that the contour
                                                                   elements are described in subprograms 1 and 2
N60 G57 P01 –2 P02 –15 P03 –8 P04 100 P05 +0
P06 +0 P07 500 * ...................................................... Cycle definition ROUGH-OUT
N70 G00 G40 G90 Z+100 M06 * ............................... Retract the spindle, insert the tool
N80 X+50 Y+50 M03 * ............................................. Pre-positioning in the X/Y plane, spindle ON
N90 Z+2 M99 * ......................................................... Pre-positioning in Z to setup clearance, cycle call
N100 Z+100 M02 *

N110 G98 L1 *
   .
   .
   .
N140 G98 L0 *
N150 G98 L2 *
   .
   .
   .
N180 G98 L0 *
N9999 %S820I G71 *
```

Subprograms: Overlapping pockets

The pocket elements A and B overlap.
The control automatically calculates the points of intersection S1 and S2, so these points do not have to be programmed.
The pockets are programmed as full circles.

```
N110    G98 L1 *                       }
N120    G01 G41 X+10 Y+50 *            }
N130    I+35 J+50 G03 X+10 Y+50 *      }  A   Left pocket
N140    G98 L0 *                       }

N150    G98 L2 *                       }
N160    G01 G41 X+90 Y+50 *            }
N170    I+65 J+50 G03 X+90 Y+50 *      }  B   Right pocket
N180    G98 L0 *                       }
N9999   % S820I G71 *
```

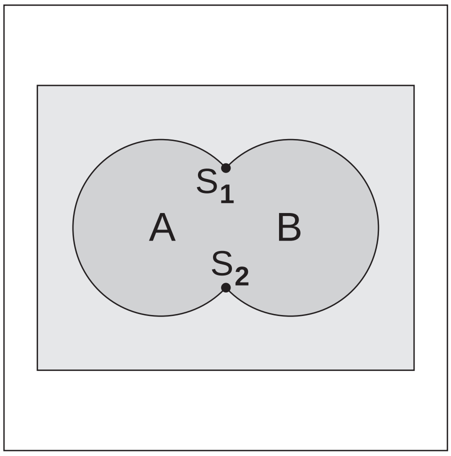


Fig. 8.17: Points of intersection S_1 and S_2 of pockets A and B

Depending on the control setup (machine parameters), machining starts either with the outline or the surface:

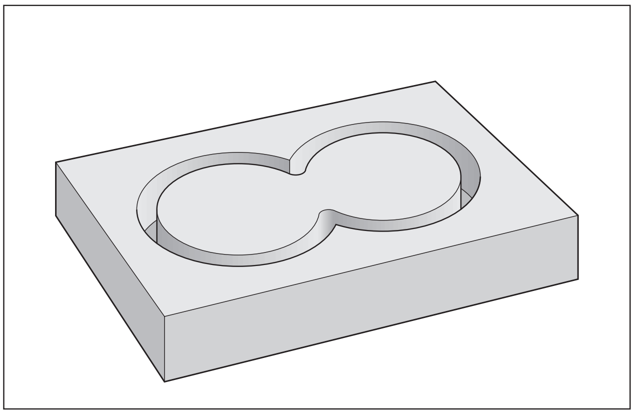


Fig. 8.18: Outline is machined first

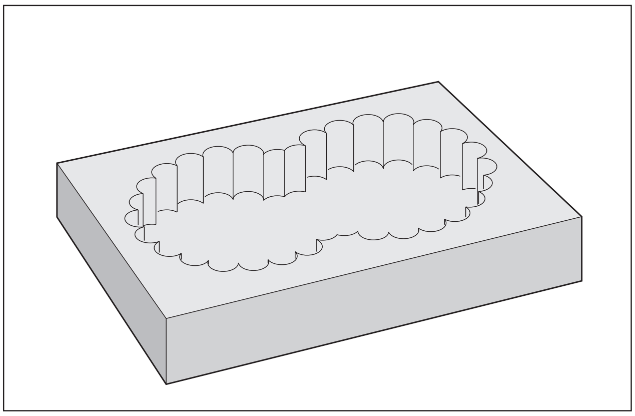


Fig. 8.19: Surface is machined first

Area of inclusion

Both areas (element A and element B) are to be machined — including the area of overlap.

- A and B must be pockets.
- The first pocket in cycle G37 must start outside the second.

```
N110    G98 L1 *
N120    G01 G41 X+10 Y+50 *
N130    I+35 J+50 G03 X+10 Y+50 *
N140    G98 L0 *

N150    G98 L2 *
N160    G01 G41 X+90 Y+50 *
N170    I+65 J+50 G03 X+50 Y+50 *
N180    G98 L0 *
```

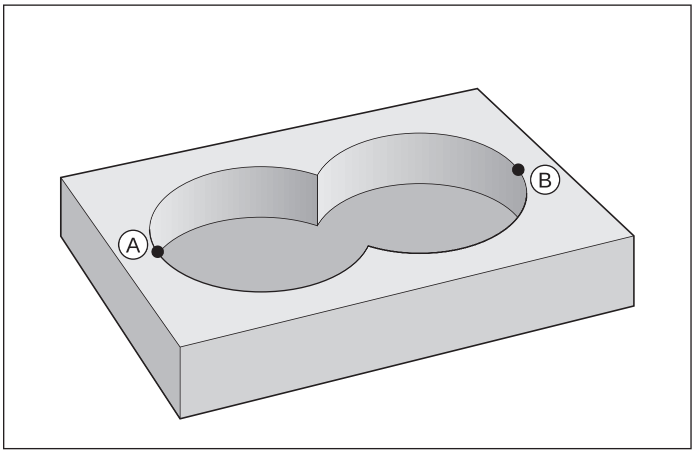


Fig. 8.20: Overlapping pockets: area of inclusion

Area of exclusion

Surface A is to be machined without the portion overlapped by B

- A must be a pocket and B an island.
- A must start outside of B.

```
N110    G98 L1 *
N120    G01 G41 X+10 Y+50 *
N130    I+35 J+50 G03 X+10 Y+50 *
N140    G98 L0 *

N150    G98 L2 *
N160    G01 G42 X+90 Y+50 *
N170    I+65 J+50 G03 X+90 Y+50 *
N180    G98 L0 *
```

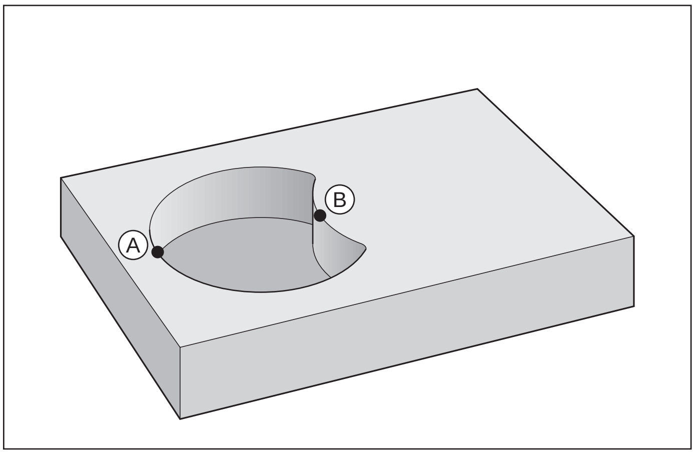


Fig. 8.21: Overlapping pockets: area of exclusion

Area of intersection

Only the area of intersection of A and B is to be machined.

- A and B must be pockets.
- A must start inside B.

```
N110    G98 L1 *
N120    G01 G41 X+60 Y+50 *
N130    I+35 J+50 G03 X+60 Y+50 *
N140    G98 L0 *

N150    G98 L2 *
N160    G01 G41 X+90 Y+50 *
N170    I+65 J+50 G03 X+90 Y+50 *
N180    G98 L0 *
```

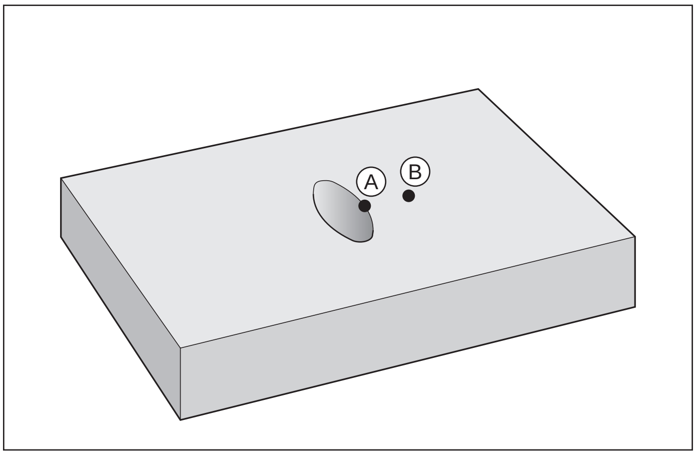


Fig. 8.22: Overlapping pockets: area of intersection

The subprograms are used in the main program on page 8-20.

Subprogram: Overlapping islands

An island always requires a pocket as an additional boundary (here, G98 L1). A pocket can also reduce several island surfaces. The starting point of this pocket must be inside the first island. The starting points of the remaining intersecting island contours must lie outside the pocket.

```
%S822I G71 *
N10     G30 G17 X+0 Y+0 Z–20 *
N20     G31 X+100 Y+100 Z+0 *
N30     G99 T1 L+0 R+2.5 *
N40     T1 G17 S2500 *
N50     G37 P01 2 P02 3 P03 1 *
N60     G57 P01 –2 P02 –10 P03 –5 P04 100
        P05 +0 P06 +0 P07 500 *
N70     G00 G40 G90 Z+100 M06 *
N80     X+50 Y+50 M03 *
N90     Z+2 M99 *
N100    Z+100 M02 *
N110    G98 L1 *
N120    G01 G41 X+5 Y+5 *
N130    X+95 *
N140    Y+95 *
N150    X+5 *
N160    Y+5 *
N170    G98 L0 *
N180    G98 L2 *
  .
  .
  .
N210    G98 L0 *
N220    G98 L3 *
  .
  .
  .
N250    G98 L0 *
N9999   %S822I G71 *
```

Area of inclusion

Elements A and B are to be left unmachined — including the area of overlap:

- A and B must be islands.
- The first island must start outside the second island.

```
N180    G98 L2 *
N190    G01 G42 X+10 Y+50 *
N200    I+35 Y+50 G03 X+10 Y+50 *
N210    G98 L0 *
N220    G98 L3 *
N230    G01 G42 X+90 Y+50 *
N240    I+65 J+50 G03 X+90 Y+50 *
N250    G98 L0 *
N9999   % S822 I G71
```

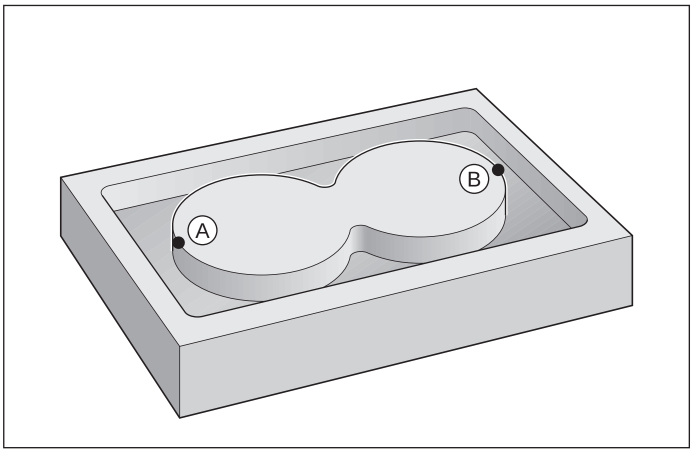


Fig. 8.23: Overlapping islands: area of inclusion

The supplements and subprograms are entered in the main program on page 8-22.

Area of exclusion

All of surface A is to be left unmachined except the portion overlapped by B:

- A must be an island and B a pocket.
- B must start inside A.

```
N180    G98 L2 *
N190    G01 G42 X+10 Y+50 *
N200    I+35 J+50 G03 X+10 Y+50 *
N210    G98 L0 *
N220    G98 L3 *
N230    G01 G41 X+40 Y+50 *
N240    I+65 J+50 G03 X+40 Y+50 *
N250    G98 L0 *
N9999   S822I G71*
```

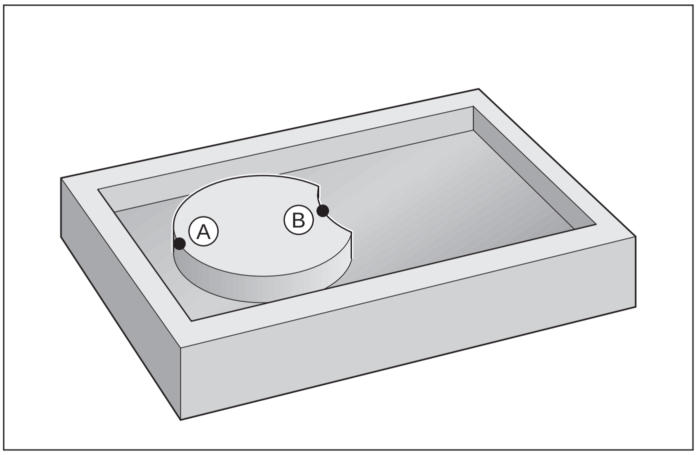


Fig. 8.24: Overlapping islands: area of exclusion

Area of intersection

Only the area of intersection of A and B is to be left unmachined.

- A and B must be islands.
- A must start inside B.

```
N180    G98 L2 *
N190    G01 G42 X+60 Y+50 *
N200    I+35 J+50 G03 X+60 Y+50 *
N210    G98 L0 *
N220    G98 L3 *
N230    G01 G42 X+90 Y+50 *
N240    I+65 J+50 G03 X+90 Y+50 *
N250    G98 L0 *
N9999   % S822I G71
```

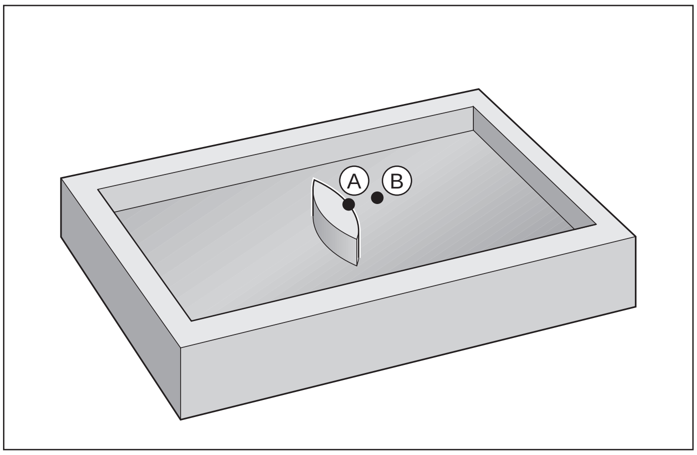


Fig. 8.25: Overlapping islands: area of intersection

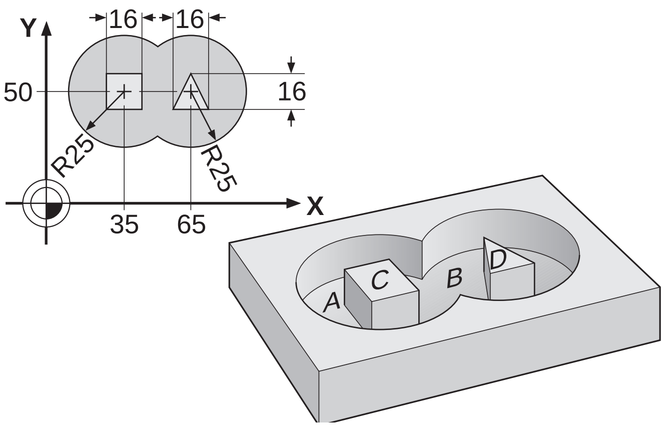
Example: Overlapping pockets and islands

PGM S824I is an expansion of PGM S820I for the inside islands C and D.

Tool: Center-cut end mill (ISO 1641),
radius 3 mm.

The contour is composed of the elements:

A and B (two overlapping pockets) as well as C and D (two islands within these pockets).



Cycle in a part program

```
%S824I G71 *
N10 G30 G17 X+0 Y+0 Z–20 *
N20 G31 X+100 Y+100 Z+0 *
N30 G99 T1 L+0 R+3 *
N40 T1 G17 S2500 *
N50 G37 P01 1 P02 2 P03 3 P04 4 *
N60 G57 P01 –2 P02 –10 P03 –5 P04 100 P05 +2 P06 +0 P07 500 *
N70 G00 G40 G90 Z+100 M06 *
N80 X+50 Y+50 M03 *
N90 Z+2 M99 *
N100 Z+100 M02 *

N110 G98 L1 *
N120 G01 G41 X+10 Y+50 *
N130 I+35 J+50 G03 X+10 Y+50 *
N140 G98 L0 *

N150 G98 L2 *
N160 G01 G41 X+90 Y+50 *
N170 I+65 J+50 G03 X+90 Y+50 *
N180 G98 L0 *

N190 G98 L3 *
N200 G01 G41 X+27 Y+42 *
N210 Y+58 *
N220 X+43 *
N230 Y+42 *
N240 X+27 *
N250 G98 L0 *

N260 G98 L4 *
N270 G01 G42 X+57 Y+42 *
N280 X+73 *
N290 X+65 Y+58 *
N300 X+57 Y+42 *
N310 G98 L0 *
N9999 %S824I G71 *
```

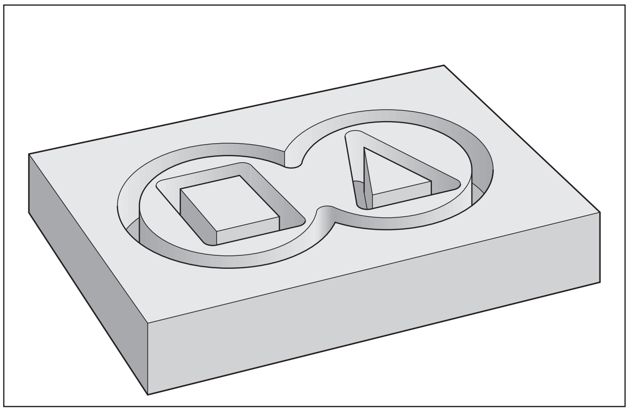


Fig. 8.26: Milling the outlines

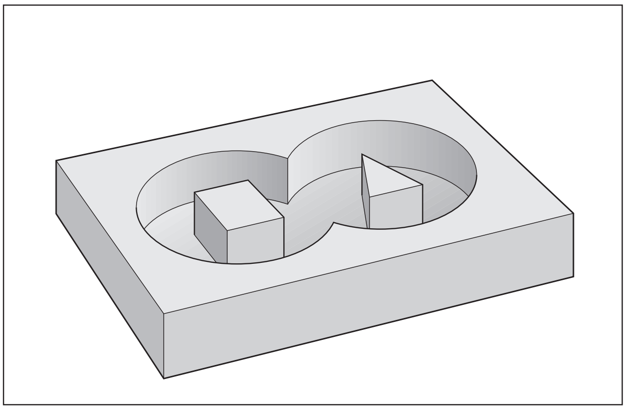


Fig. 8.27: Milling completed

PILOT DRILLING G56

Process

Pilot drilling of holes for cutter infeed at the starting points of the subcontours. With SL contours that consist of several overlapping surfaces, the cutter infeed point is the starting point of the first subcontour:

- The tool is positioned above the first infeed point.
- The subsequent drilling sequence is identical to that of cycle G83: PECKING.
- The tool is then positioned above the next infeed point, and the drilling process is repeated.

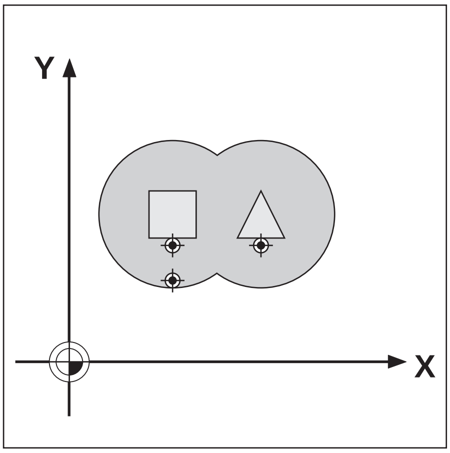


Fig. 8.28: Example of cutter infeed for PECKING

Input data

- SETUP CLEARANCE
- MILLING DEPTH
- PECKING DEPTH
- DWELL TIME
- FEED RATE

} Identical to cycle G83 PECKING

- FINISHING ALLOWANCE
  Allowed material for the drilling operation (see Fig. 8.29).
  The sum of tool radius and finishing allowance should be the same for pilot drilling and roughing out.

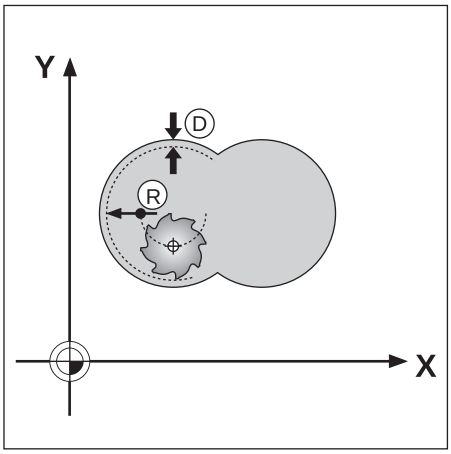


Fig. 8.29: Finishing allowance

CONTOUR MILLING G58/G59

Cycles G58/G59 are used to finish-mill the contour pocket.
This cycle can also be used generally for milling contours.

Process

- The tool is positioned above the first starting point.
- The tool then penetrates at the programmed feed rate to the first pecking depth.
- On reaching the first pecking depth, the tool mills the first contour at the programmed feed rate and in the specified direction of rotation.
- At the infeed point, the tool is advanced to the next pecking depth.

This process is repeated until the programmed milling depth is reached. The remaining subcontours are milled in the same manner.

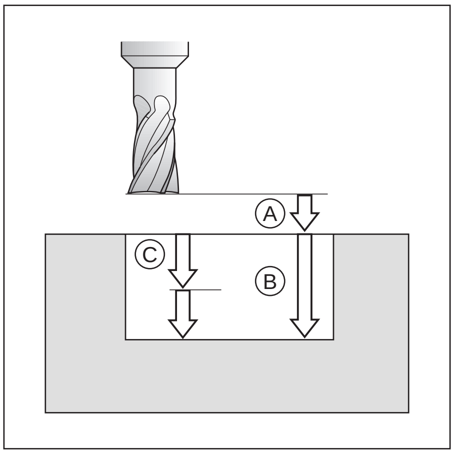


Fig. 8.30: Infeeds and distances for CONTOUR MILLING

Required tool

This cycle requires a center-cut end mill (ISO 1641).

Direction of rotation for CONTOUR MILLING

With clockwise direction of rotation G58:

- M3 defines climb milling for pocket and island.

With counterclockwise direction of rotation G59:

- M3 defines up-cut milling for pocket and island.

Input data

- SETUP CLEARANCE Ⓐ
- MILLING DEPTH Ⓑ
- PECKING DEPTH Ⓒ
- FEED RATE FOR PECKING:
  Traversing speed of the tool during penetration.
- FEED RATE:
  Traversing speed of the tool in the working plane.

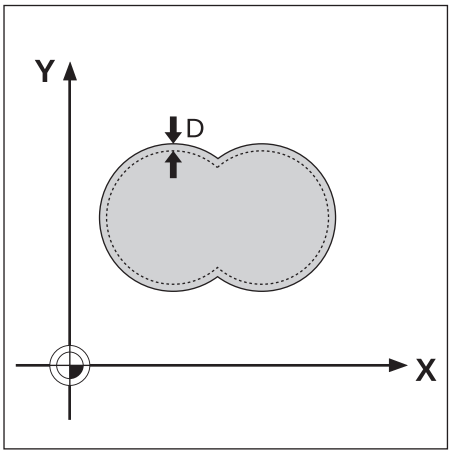


Fig. 8.31: Finishing allowance

The following scheme illustrates the application of the cycles Pilot Drilling, Rough-Out and Contour Milling in part programming:

1. List of contour subprograms

G37
Cycle call not required.

2. Drilling

Define and call drilling tool
G56
Pre-positioning
Cycle call required!

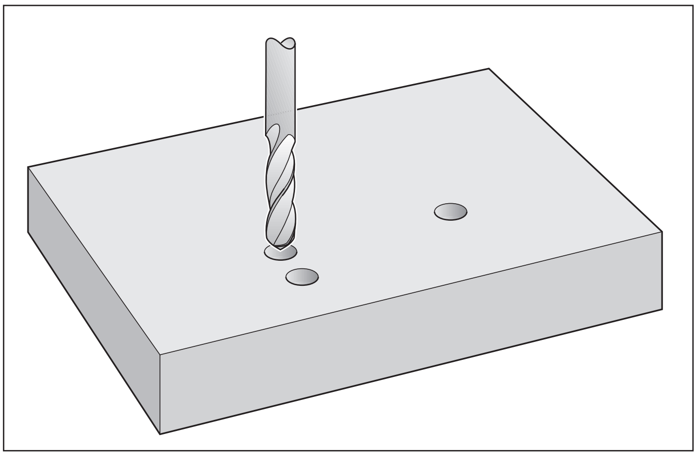


Fig. 8.32: PILOT DRILLING cycle

3. Rough-out

Define and call tool for rough milling
G57
Pre-positioning
Cycle call required!

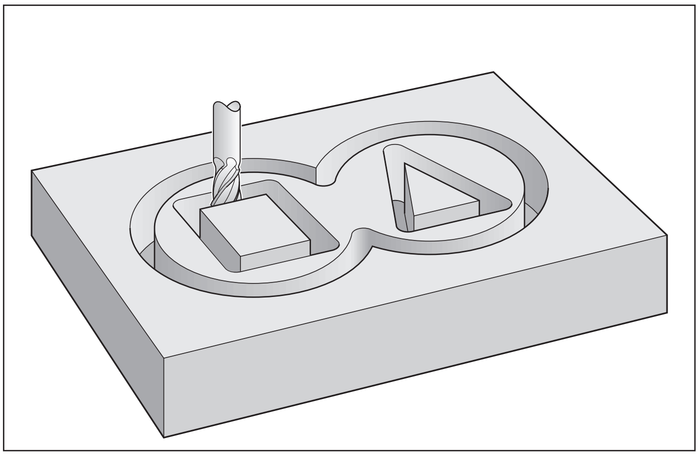


Fig. 8.33: ROUGH-OUT cycle

4. Finishing

Define and call finish milling tool
G58/G59
Pre-positioning
Cycle call required!

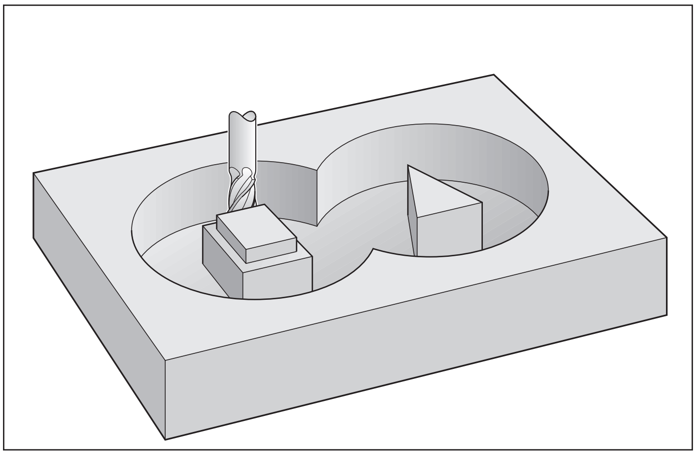


Fig. 8.34: CONTOUR MILLING cycle

5. Contour subprograms

M02 *
Subprograms for the subcontours.

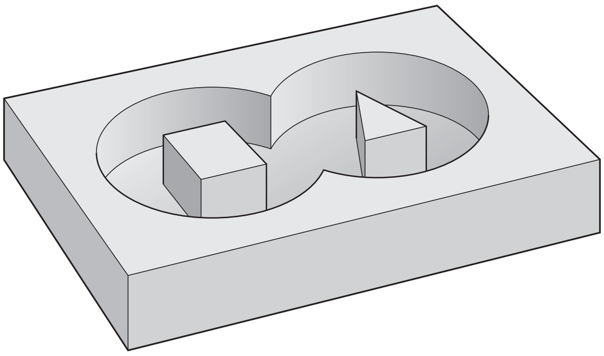
Example: Overlapping pockets with islands

Inside machining with pilot drilling, roughing out and finishing.

PGM S829I is based on S824I:

The main program has been expanded by the cycle definitions and cycle calls for pilot drilling and finishing.

The contour subprograms 1 to 4 are identical to those in PGM S824I (see page 8-24) and are added after block N300.



```
%S829I G71 * ............................................ Begin program
N10 G30 G17 X+0 Y+0 Z–20 * .............................. Define workpiece blank
N20 G31 X+100 Y+100 Z+0 *
N30 G99 T1 L+0 R+2.5 * .................................. Tool definition drill
N40 G99 T2 L+0 R+3 * .................................... Tool definition rough mill
N50 G99 T3 L+0 R+2.5 * .................................. Tool definition finish mill
N60 L10,0 * ............................................. Subprogram call for tool change
N70 G38 M06 * ........................................... Stop program run
N80 T1 G17 S2500 * ...................................... Tool call drill
N90 G37 P01 1 P02 2 P03 3 P04 4 * ....................... Cycle definition CONTOUR GEOMETRY
N100 G56 P01 –2 P02 –10 P03 –5 P04 500 P05 +2 * ......... Cycle definition PILOT DRILLING
N110 Z+2 M03 *
N120 G79 * .............................................. Cycle call PILOT DRILLING
N130 L10,0 *
N140 G38 M06 * .......................................... Tool change
N150 T2 G17 S1750 * ..................................... Tool call rough mill
N160 G57 P01 –2 P02 –10 P03 –5 P04 100 P05+2
P06+0 P07 500 * ......................................... Cycle definition ROUGH-OUT
N170 Z+2 M03 *
N180 G79 * .............................................. Cycle call ROUGH-OUT
N190 L10,0 *
N200 G38 M06 * .......................................... Tool change
N210 T3 G17 S2500 * ..................................... Tool call finish mill
N220 G58 P01 –2 P02 –10 P03 –10 P04 100
P05 500 * ............................................... Cycle definition CONTOUR MILLING
N230 Z+2 M03 *
N240 G79 * .............................................. Cycle call CONTOUR MILLING
N250 Z+100 M02 *
N260 G98 L10 * .......................................... Subprogram for tool change
N270 T0 G17 *
N280 G00 G40 G90 Z+100 *
N290 X–20 Y–20 *
N300 G98 L0 *
```

From block N310: add the subprograms listed on page 8-24

```
N9999 %S829I G71 *
```

8.4 Cycles for Coordinate Transformations

Coordinate transformations enable a programmed contour to be changed in its position, orientation or size. A contour can be:

- shifted (DATUM SHIFT cycles G53/G54)
- mirrored (MIRROR IMAGE cycle G28)
- rotated (ROTATION cycle G73)
- made smaller or larger
  (SCALING FACTOR cycle G72)

The original contour must be identified as a subprogram or program section.

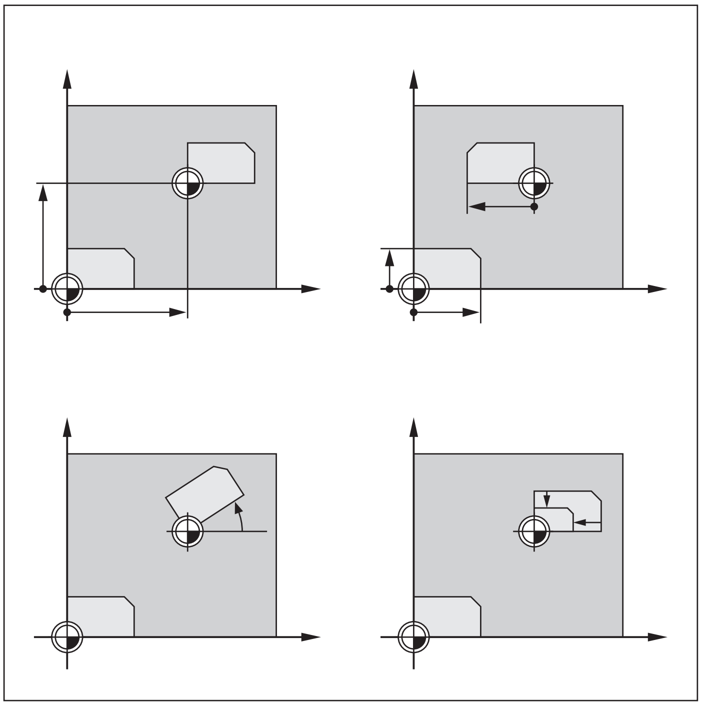


Fig. 8.35: Examples of coordinate transformations

Activation of coordinate transformations

Immediate activation: A coordinate transformation becomes effective as soon as it is defined (it does not have to be called). The transformation remains effective until it is changed or cancelled.

To cancel a coordinate transformation:

- Define cycle for basic behavior with new values, for example scaling factor 1.
- Execute miscellaneous function M02, M30 or block N 9999 % ... (depending on machine parameters).
- Select a new program.

DATUM SHIFT G54

Application

You can repeat machining operations at various locations on the workpiece by shifting the datum.

Activation

When the DATUM SHIFT cycle has been defined, all coordinate data are based on the new datum. Shifted axes are identified in the status display.

Input data

Only the coordinates of the new datum need to be entered. Absolute values are based on the workpiece datum manually defined with datum setting. Incremental values are based on the last valid datum; this datum can itself be shifted.

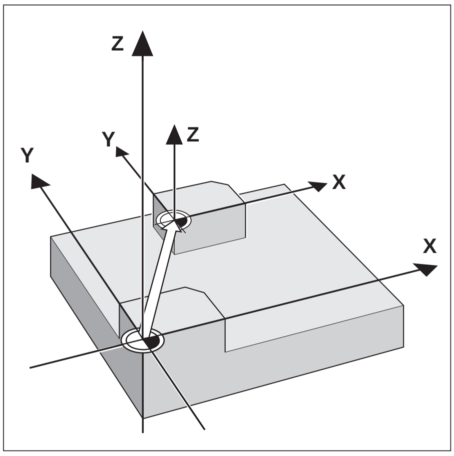


Fig. 8.36: Activation of the datum shift

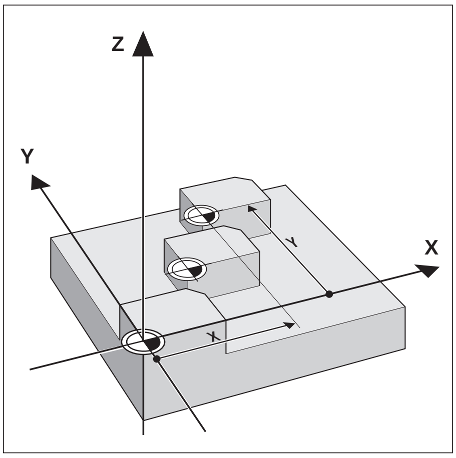


Fig. 8.37: Datum shift, absolute

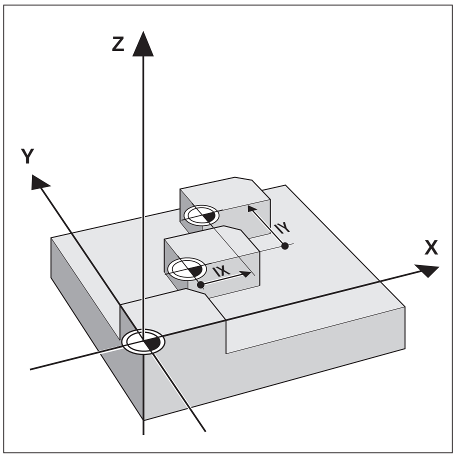


Fig. 8.38: Datum shift, incremental

Cancellation

To cancel a datum shift, enter the datum shift coordinates X = 0, Y = 0 and Z = 0.

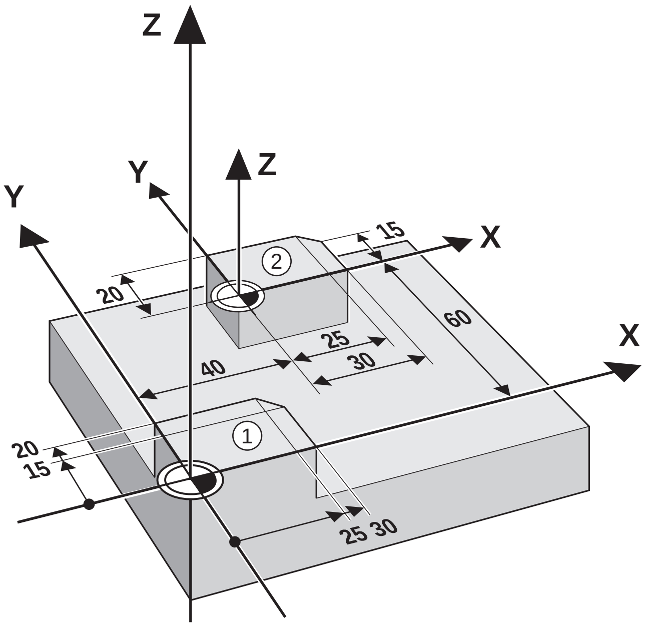
When combining transformations, program the datum shift first.

Example: Datum shift

A machining sequence in the form of a subprogram is to be executed twice:

a) once, referenced to the specified datum ① X+0/Y+0 and

b) a second time, referenced to the shifted datum ② X+40/Y+60.



Cycle in a part program

```
%S840I G71 * ............................................................ Begin program
N10 G30 G17 X+0 Y+0 Z–20 * ................................... Define workpiece blank
N20 G31 X+100 Y+100 Z+0 *
N30 G99 T1 L+0 R+4 * .............................................. Tool definition
N40 T1 G17 S1500 * .................................................. Tool call
N50 G00 G40 G90 Z+100 * ........................................ Retract the tool
N60 L1,0 * .................................................................. Execute sequence 1 without datum shift
N70 G54 X+40 Y+60 *
N80 L1,0 * .................................................................. Execute sequence 2 with datum shift
N90 G54 X+0 Y+0 * ................................................... Cancellation of datum shift
N100 Z+100 M02 *
N110 G98 L1 *
        .
        .
        .
N230 G98 L0 *
N9999 %S840I G71 *
```

Subprogram:

```
N110 G98 L1 *
N120 X–10 Y–10 M03 *
N130 Z+2 *
N140 G01 Z–5 F200 *
N150 G41 X+0 Y+0 *
N160 Y+20 *
N170 X+25 *
N180 X+30 Y+15 *
N190 Y+0 *
N200 X+0 *
N210 G40 X–10 Y–10 *
N220 G00 Z+2 *
N230 G98 L0 *
```

The location of the subprogram (NC block) depends on the transformation cycle:

	LBL 1	**LBL 0**
Datum shift	block N110	block N230
Mirror image, rotation, scaling	block N130	block N250

MIRROR IMAGE G28

Application

This cycle makes it possible to machine the mirror image of a contour in the working plane.

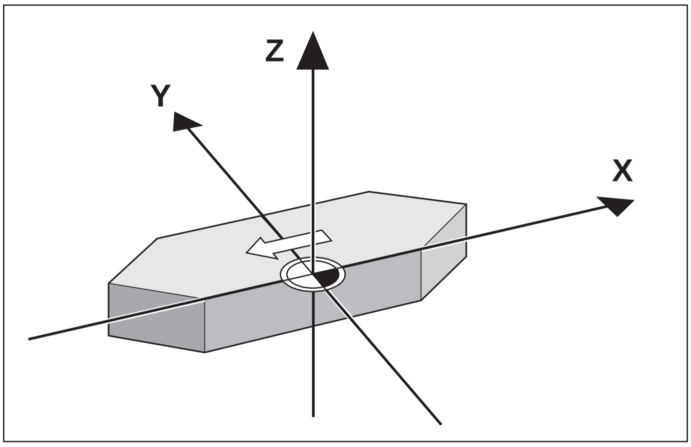


Fig. 8.39: MIRROR IMAGE cycle

Activation

The MIRROR IMAGE cycle becomes active as soon as it is defined. Mirrored axes are identified in the status display.

- If one axis is mirrored, the machining direction of the tool is reversed. This does not apply to fixed cycles, however.
- If two axes are mirrored, the machining direction remains the same.

The mirror image depends on the location of the datum:

- If the datum is located on the mirrored contour, the part "turns over" at that point.
- If the datum is located outside the mirrored contour, the part turns over and also "jumps" to another location.

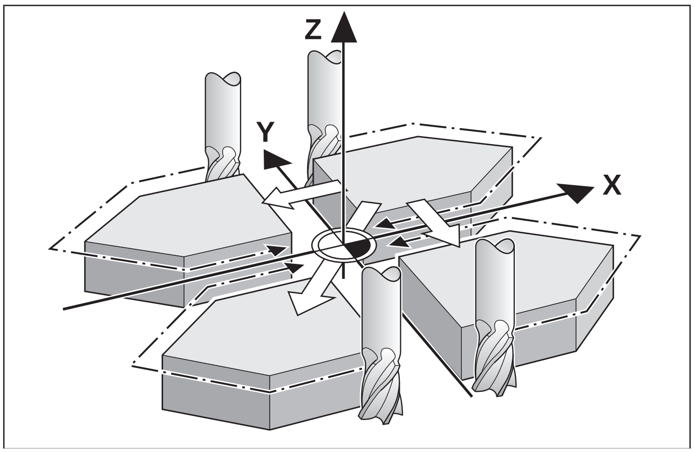


Fig. 8.40: Multiple mirroring and milling direction

Input data

Enter the axis that you wish to mirror. The tool axis cannot be mirrored.

Cancellation

To cancel the mirror image, program G28 without defining an axis.

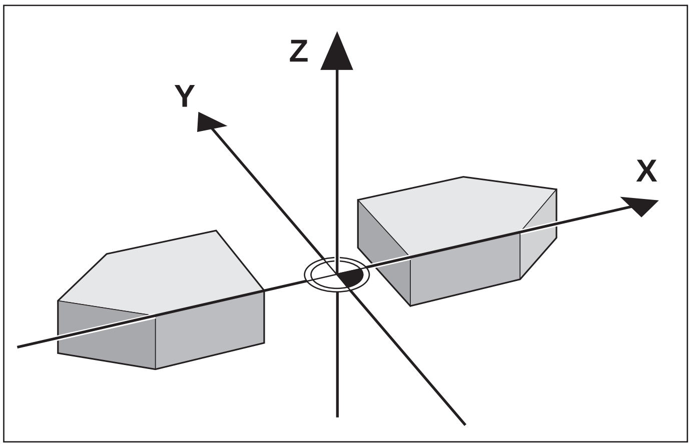
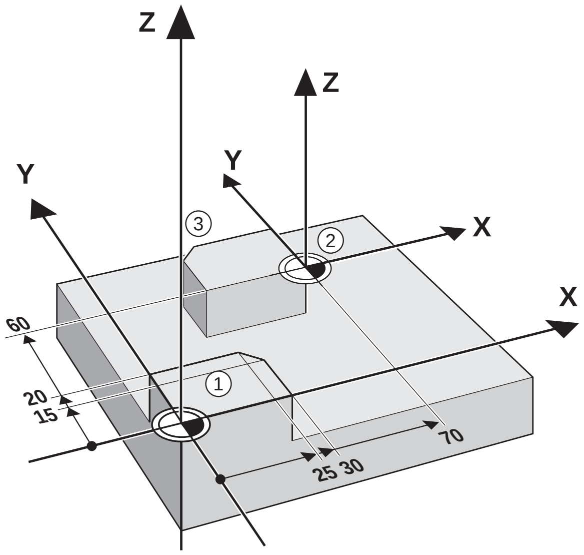


Fig. 8.41: Datum lies outside the mirrored contour

Example: Mirror Image

A machining sequence (subprogram 1) is to be executed once – as originally programmed – referenced to the datum X+0/Y+0 ① and then again referenced to X+70/Y+60 ② mirrored ③ in X.



MIRROR IMAGE cycle in a part program

```
%S844I G71 * ............................................................Begin program
N10 G30 G17 X+0 Y+0 Z–20 * ..................................Define workpiece blank
N20 G31 X+100 Y+100 Z+0 *
N30 G99 T1 L+0 R+4 * .............................................Tool definition
N40 T1 G17 S1500 * ..................................................Tool call
N50 G00 G40 G90 Z+100 * .......................................Retract the tool
N60 L1,0 * .................................................................Execute sequence 1, not mirrored
N70 G54 X+70 Y+60 * ...............................................Datum shift
N80 G28 X * ..............................................................Activate mirror image
N90 L1,0 * .................................................................Execute sequence 2 with datum shift and mirror image
N100 G28 * ................................................................Cancel mirror image
N110 G54 X+0 Y+0 * .................................................Cancel datum shift
N120 Z+100 M02 *
N130 G98 L1 *                                    }
  .                                              } The subprogram is identical to the subpro-
  .                                              } gram shown on page 8-32
  .                                              }
N250 G98 L0 *
N9999 %S844I G71 *
```

ROTATION G73

Application

Within a program the coordinate system can be rotated around the active datum in the working plane.

Activation

A rotation becomes active as soon as the cycle is defined. This cycle is also effective in the POSITIONING WITH MANUAL INPUT mode.

Reference axis for the rotation angle:

• X/Y plane	X axis
• Y/Z plane	Y axis
• Z/X plane	Z axis

The active rotation angle is indicated in the status display.

Parallel axes U,V,W cannot be rotated.

Input data

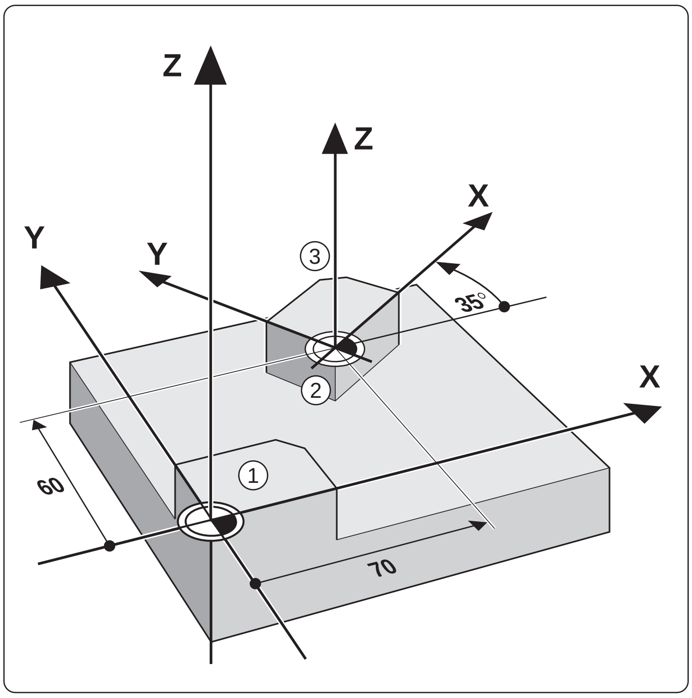
The angle of rotation is entered in degrees (°).
Entry range: –360° to +360° (absolute or incremental)

Cancellation

To cancel a rotation, enter a rotation angle of 0°.

Example: Rotation

A contour (subprogram 1) is to be executed once – as originally programmed – referenced to the datum X+0/Y+0 and then executed again referenced to X+70 Y+60 and rotated by 35°.



Continued...

Cycle in a part program

```
%S846I G71 * ........................................................ Begin program
N10 G30 G17 X+0 Y+0 Z–20 * ...................................... Define workpiece blank
N20 G31 X+100 Y+100 Z+0 *
N30 G99 T1 L+0 R+4 * ............................................ Tool definition
N40 T1 G17 S1500 * .............................................. Tool call
N50 G00 G40 G90 Z+100 * ......................................... Retract the tool
N60 L1,0 * ...................................................... Execute sequence 1, non-rotated
N70 G54 X+70 Y+60 *
N80 G73 G90 H+35 *
N90 L1,0 * ...................................................... Execute sequence 2 with datum shift and rotation
N100 G73 G90 H+0 * .............................................. Cancel rotation
N110 G54 X+0 Y+0 * .............................................. Cancel datum shift
N120 Z+100 M02 *

N130 G98 L1 *                                                  }
   .                                                           }
   .                                                           } This subprogram is identical to the subpro-
   .                                                           } gram on page 8-32
                                                               }
N250 G98 L0 *                                                  }
N9999 %S846I G71 *
```

The corresponding subprogram (see page 8-32) is programmed after M2.

SCALING FACTOR G72

Application

This cycle allows you to increase or reduce the size of contours within a program, such as for shrinkage or finishing allowances.

Activation

A scaling factor becomes effective as soon as the cycle is defined. Scaling factors can be applied

- in the working plane or to all three coordinate axes at the same time (depending on MP7410)
- to the dimensions in cycles
- also in the parallel axes U, V, W.

Input data

The cycle is defined by entering the scaling factor F. The TNC multiplies the coordinates and radii by the F factor (as described under "Activation" above).

To increase the size:	enter F greater than 1 (max. 99.999 999)
To reduce the size:	enter F smaller than 1 (down to 0.000 001)

Cancellation

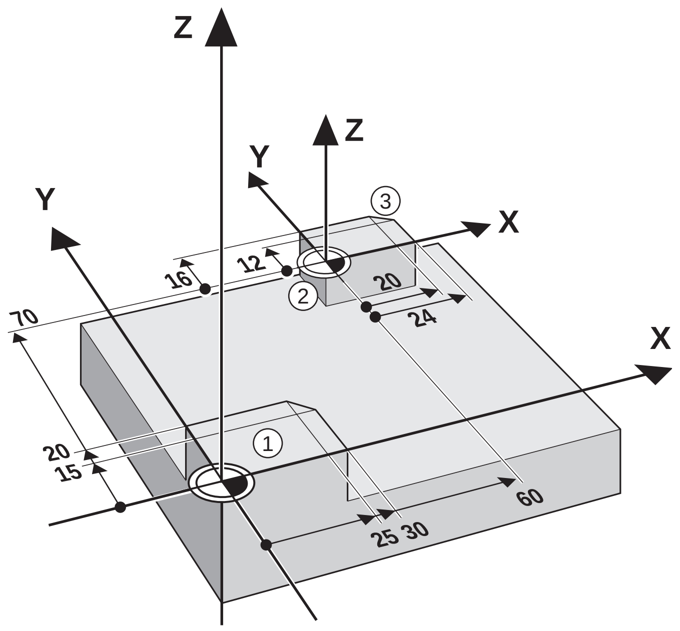
To cancel a scaling factor, enter a scaling factor of 1.

Prerequisite

Before entering a scaling factor it is advisable to set the datum to an edge or corner of the contour.

Example: Scaling factor

A contour (subprogram 1) is to be executed once – as originally programmed – referenced to the manually set datum X+0/Y+0 and then executed again referenced to X+60/Y+70 and reduced by a scaling factor of 0.8.



SCALING FACTOR cycle in a part program

```
%S847I G71 * ............................................ Begin program
N10 G30 G17 X+0 Y+0 Z–20 * .............................. Define workpiece blank
N20 G31 X+100 Y+100 Z+0 *
N30 G99 T1 L+0 R+4 * .................................... Tool definition
N40 T1 G17 S1500 * ...................................... Tool call
N50 G00 G40 G90 Z+100 * ................................. Retract the tool
N60 L1,0 * .............................................. Execute sequence 1 at original size
N70 G54 X+70 Y+60 *
N80 G72 F0.8 *
N90 L1,0 * .............................................. Execute sequence 2 with datum shift and scaling factor
N100 G72 F1 * ........................................... Cancel scaling factor
N110 G54 X+0 Y+0 * ...................................... Cancel datum shift
N120 Z+100 M02 *

N130 G98 L1 *            }
  .                      }
  .                      } This subprogram is identical to the subpro-
  .                      } gram shown on page 8-32
N250 G98 L0 *            }
N9999 %S847I G71 *
```

The corresponding subprogram (see page 8-32) is programmed after M2.

8.5 Other Cycles

DWELL TIME G04

Application

Within a running program, the execution of the next block is delayed by the programmed dwell time.

A dwell time cycle can be used, for example, for chip breaking.

Activation

This cycle becomes effective as soon as it is defined. Modal conditions (such as spindle rotation) are not affected.

Input data

The dwell time is programmed with G04 followed by F and the desired dwell time in seconds.
Entry range: 0 to 30 000 s (approx. 8.3 hours) in increments of 0.001 s.

*Example NC block: N135 G04 F3 **

PROGRAM CALL G39

Application and activation

Part programs such as special drilling cycles, curve milling or geometric modules, can be written as main programs and then called for use just like fixed cycles.

Input data

Enter the file name of the program to be called.

The program is called with

- G79 (separate block) or
- M99 (blockwise) or
- M89 (modally)

Example: Program call

A callable program (program 50) is to be called into a program with a cycle call.

Part program

```
  .
  .
  .
G39 P01 50 ........................................................ Definition: "Program 50 is a cycle"
G00 G40 X+20 Y+50 M99 ..................................... Call of program 50
  .
  .
  .
```

ORIENTED SPINDLE STOP G36

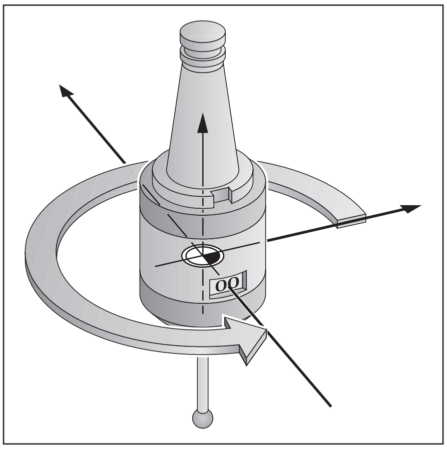


Fig. 8.42: Oriented spindle stop

Application

The TNC can address the machine tool spindle as a 5th axis and turn it to a certain angular position. Oriented spindle stops are required for:

- Tool changing systems with a defined tool change position
- Orientation of the transmitter/receiver window of the TS 511 Touch Probe System from HEIDENHAIN.

Activation

The angle of orientation defined in the cycle is positioned to with M19.
If M19 is executed without a cycle definition, the machine tool spindle will be oriented to the angle set in the machine parameters.

Oriented spindle stops can also be programmed in machine parameters.

Prerequisite

The machine must be set up for this cycle.

Input data

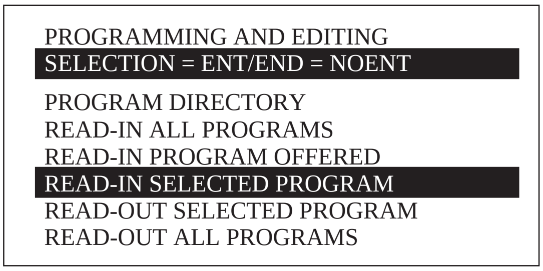
Angle of orientation S (based on the angle reference axis of the working plane)

Input range:	0 to 360°.
Input resolution:	0.1°.

The TNC features an RS-232-C data interface for transferring data to and from other devices. It can be used in the PROGRAMMING AND EDITING operating mode and in a program run mode.

Possible applications:

- Blockwise transfer (DNC mode)
- Downloading program files into the TNC
- Transferring program files from the TNC to external storage devices
- Printing files



PROGRAMMING AND EDITING
SELECTION = ENT/END = NOENT

PROGRAM DIRECTORY
READ-IN ALL PROGRAMS
READ-IN PROGRAM OFFERED
READ-IN SELECTED PROGRAM
READ-OUT SELECTED PROGRAM
READ-OUT ALL PROGRAMS

Fig. 9.1: Menu for external data transfer

9.1 Menu for External Data Transfer

To select external data transfer:

EXT	Menu for external data transfer appears on the screen.

Use the arrow keys to select the individual menu options.

Function	**Menu option**
Display program numbers of the programs on the storage medium	PROGRAM DIRECTORY
Transfer all programs from the storage medium into the TNC	READ-IN ALL PROGRAMS
Display programs for transfer into the TNC	READ-IN PROGRAM OFFERED
Transfer selected program into the TNC	READ-IN SELECTED PROGRAM
Transfer selected program to an external device	READ-OUT SELECTED PROGRAM
Transfer all programs which are in TNC memory to an external device	READ-OUT ALL PROGRAMS

Aborting data transfer

To abort a data transfer process, press END.

If you are transferring data between two TNCs, the receiving control must be started first.

Blockwise transfer

In the operating modes PROGRAM RUN/FULL SEQUENCE and SINGLE BLOCK, it is possible to transfer programs which exceed the memory capacity of the TNC by means of blockwise transfer with simultaneous execution (see page 3-6).

9.2 Pin Layout and Connecting Cable for Data Interfaces

RS-232-C/V.24 Interface

HEIDENHAIN devices

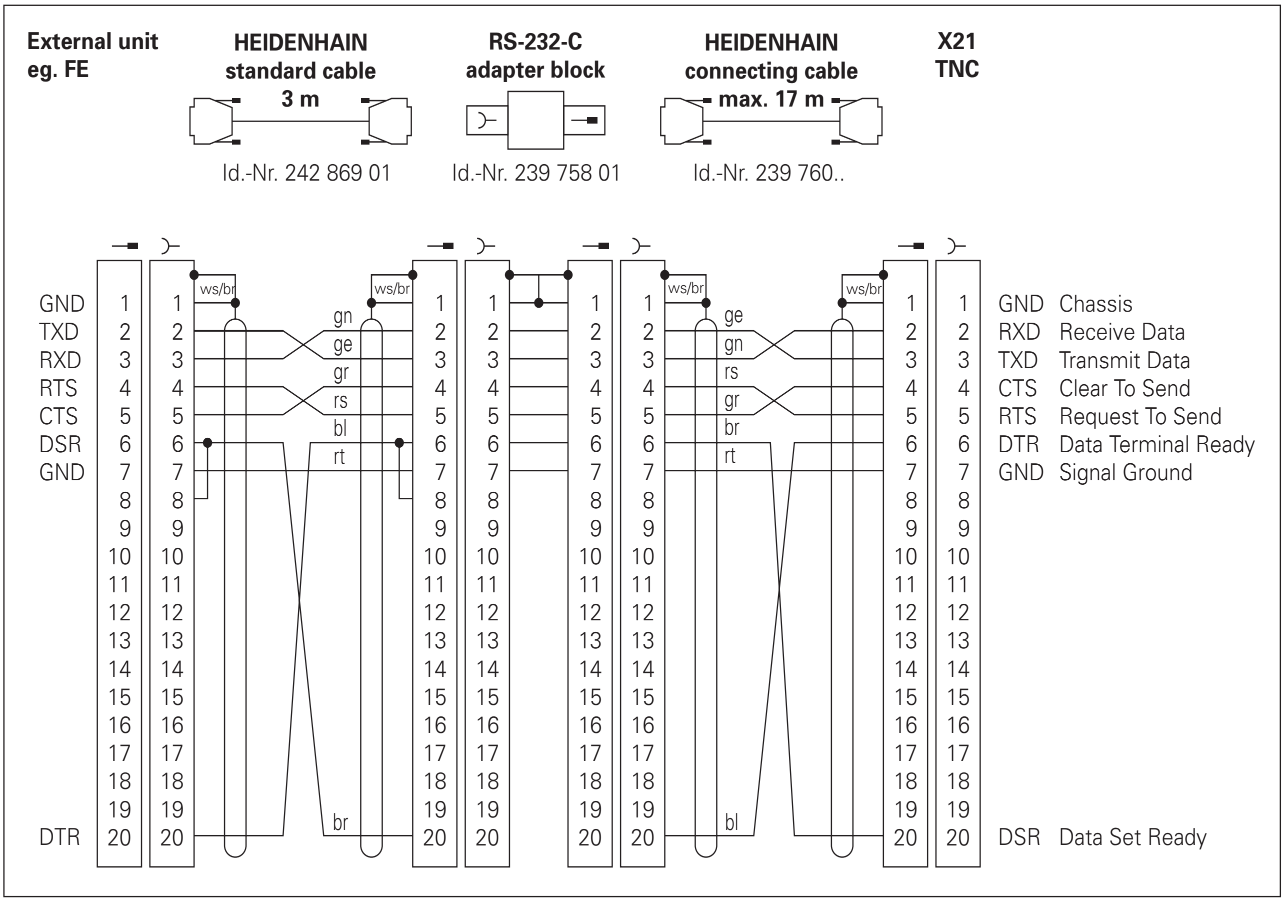


Fig. 9.2: Pin layout of the RS-232-C/V.24 interface for HEIDENHAIN devices

The connecting pin layout on the TNC logic unit (X25) is different from that on the adapter block.

Non-HEIDENHAIN devices

The connector pin layout on a non-HEIDENHAIN device may differ considerably from that on a HEIDENHAIN device. The pin layout will depend on the unit and the type of data transfer.

9.3 Preparing the Devices for Data Transfer

HEIDENHAIN devices

HEIDENHAIN devices (FE floppy disk unit and ME magnetic tape unit) are designed for use with the TNC. They can be used for data transfer without further adjustments.

Example: FE 401 Floppy Disk Unit

- Connect the power cable to the FE
- Connect the FE and the TNC with data transfer cable
- Switch on the FE
- Insert a diskette into the upper drive
- Format the diskette if necessary
- Set the interface (see page 10-3)
- Transfer the data

The baud rate can be selected on the FE 401 floppy disk unit.

Non-HEIDENHAIN devices

The TNC and non-HEIDENHAIN devices must be adapted to each other.

Adapting a non-HEIDENHAIN device for the TNC

- PC: Adapt the software
- Printer: Adjust the DIP switches

Adapting the TNC for a non-HEIDENHAIN device

- Set user parameter 5020.

The MOD functions provide additional displays and input possibilities. The MOD functions available depend on the selected operating mode.

Functions available in the operating modes PROGRAMMING AND EDITING and TEST RUN:

- Display NC software number
- Display PLC software number
- Enter code number
- Set the data interface
- Machine-specific user parameters

Functions available in all other modes:

- Display NC software number
- Display PLC software number
- Select position display
- Select unit of measurement (mm/inch)
- Select programming language
- Set traverse limits

10.1 Selecting, Changing and Exiting the MOD Functions

To select the MOD functions:

MOD	Select the MOD functions.

To change the MOD functions:

Select the desired MOD function with the arrow keys.

↑ / ↓ ENT Repeatedly	Page through the MOD functions until you find the desired function.

To exit the MOD functions:

END □	Close the MOD functions.

10.2 NC and PLC Software Numbers

The software numbers of the NC and PLC are displayed in the dialog field when the corresponding MOD function is selected.

10.3 Entering the Code Number

The TNC asks for a code number before allowing access to certain functions:

Function	Code number
Cancel erase/edit protection (status P)	86357
Select user parameters	123
Timers for: Control ON Program run Spindle ON	857282

Code numbers are entered in the dialog field after the corresponding MOD function is selected.

10.4 Setting the External Data Interfaces

Two functions are available for setting the external data interface:

- BAUD RATE
- RS-232 INTERFACE

Use the vertical arrow keys to select the functions.

BAUD RATE

The baud rate is the speed of data transfer in bits per second.

Permissible baud rates (enter with the numerical keys):
110, 150, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400 baud

The ME 101 has a baud rate of 2400.

RS-232-C Interface

The proper setting depends on the connected device.

Use the ENT key to select the baud rate.

External device	RS-232-C interface =
HEIDENHAIN FE 401 and FE 401B floppy disk units	FE
HEIDENHAIN ME 101 magnetic tape unit (no longer in production)	ME
Non-HEIDENHAIN units such as printers, tape punchers, and PCs without TNC.EXE	EXT
No transfer of data, e.g. digitizing without transfer of the digitized data or operation without connecting a device	– empty –

10.5 Machine-Specific User Parameters

The machine tool builder can assign functions to up to 16 USER PARAMETERS. For more detailed information, refer to the operating manual for the machine tool.

10.6 Selecting Position Display Types

The positions indicated in Fig. 10.1 are:

- Starting position Ⓐ
- Target position of the tool Ⓩ
- Workpiece datum Ⓦ
- Scale datum Ⓜ

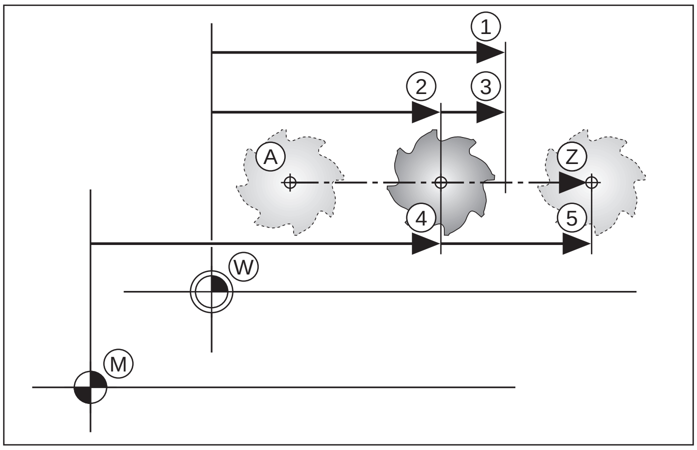


Fig. 10.1: Characteristic positions on the workpiece and scale

The TNC position display can show the following coordinates:

- Nominal position (the value presently commanded by the TNC) ① .... NOML.
- Actual position (the position at which the tool is presently located) ② .... ACTL.
- Servo lag (difference between the nominal and actual positions) ③ .... LAG
- Reference position (the actual position as referenced to the scale datum) ④ .... REF
- Distance remaining to the programmed position (difference between actual and target position) ⑤ .... DIST.

Select the desired information with the ENT key. It is then displayed directly in the status field.

10.7 Selecting the Unit of Measurement

This MOD function determines whether coordinates are displayed in millimeters or inches.

- Metric system: e.g. X = 15.789 (mm)
  MOD function CHANGE MM/INCH
  The value is displayed with 3 places after the decimal point
- Inch system: e.g. X = 0.6216 (inch)
  MOD function CHANGE MM/INCH
  The value is displayed with 4 places after the decimal point

10.8 Selecting the Programming Language

The MOD function PROGRAM INPUT lets you choose between programming in HEIDENHAIN plain language format and ISO format:

- To program in HEIDENHAIN format:
  Set the PROGRAM INPUT function to HEIDENHAIN
- To program in ISO format:
  Set the PROGRAM INPUT function to: ISO

10.9 Setting the Axis Traverse Limits

The MOD function AXIS LIMIT allows you to set limits to axis traverse within the machine's maximum working envelope.

Possible application:
to protect an indexing fixture from tool collision.

The maximum traverse range is defined by software limit switches. This range can be additionally limited through the MOD function AXIS LIMIT. With this function you can enter the maximum traverse positions for the positive and negative directions. These values are referenced to the scale datum.

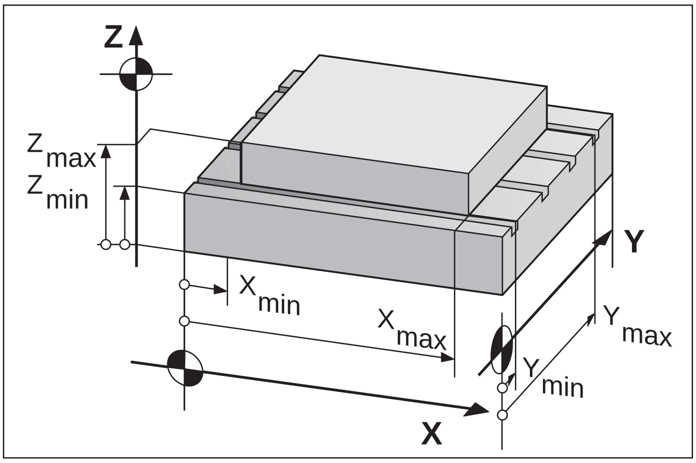


Fig. 10.2: Traverse limits on the workpiece

Working without additional traverse limits

To allow certain coordinate axes to use their full range of traverse, enter the maximum traverse of the TNC (+/–30 000 mm) as the AXIS LIMIT.

To find and enter the maximum traverse:

Select POSITION DISPLAY REF.

Move the spindle to the desired positive and negative end positions of the X, Y and Z axes.

Write down the values, noting the algebraic sign.

MOD — Select the MOD functions.

Enter the values that you wrote down as LIMITS in the corresponding axes.

END □ — Exit the MOD functions.

- The tool radius is not automatically compensated in the axis traverse limits values.
- Traverse range limits and software limit switches become active as soon as the reference marks are crossed over.
- In every axis the TNC checks whether the negative limit is smaller than the positive one.
- The reference positions can also be captured directly with the function "Actual Position Capture" (see page 4-20).

11.1 General User Parameters

General user parameters are machine parameters which affect the behavior of the TNC. These parameters set such things as:

- Dialog language
- Interface behavior
- Traversing speeds
- Machining sequences
- Effect of the overrides

Selecting the general user parameters

General user parameters are selected with code number 123 in the MOD functions.

MOD functions also include machine-specific user parameters.

Parameters for external data transfer

These parameters define control characters for blockwise transfer.

Input values: Number between 0 and 32 382
(ASCII character with 16-bit coding)

Note:
The character defined here for end of program is also valid for the setting of the standard interface.

MP5010

Function	MP	Bit
• End of program	5010.0	0 to 7
• Beginning of program	5010.0	8 to 15
• Data input	5010.1	0 to 15
• Data output	5010.2	0 to 15
• Beginning of command block	5010.3	0 to 7
• End of command block	5010.3	8 to 15
• Positive acknowledgment	5010.4	0 to 7
• Negative acknowledgment	5010.4	8 to 15
• End of data transfer	5010.5	0 to 15

Integrating the TNC interfaces to external devices: Data format and transmission stop

Input value: number between 0 and 255
The entry value is the sum of the individual values.

MP5020

Function	Selections	Value
• Number of data bits	7 data bits (ASCII code, 8th bit = parity)	+0
	8 data bits (ASCII code, 9th bit = parity)	+1
• Block Check Character (BCC)	BCC can be any character	+0
	BCC control character not allowed	+2
• Transmission stop with RTS	Active	+4
	Inactive	+0
• Transmission stop with DC3	Active	+8
	Inactive	+0
• Character parity	Even	+0
	Odd	+16
• Character parity	Not desired	+0
	Desired	+32
• Number of stop bits	$1^1/_2$ stop bits	+0
	2 stop bits	+64
	1 stop bit	+128
	1 stop bit	+192

Example

To adapt the TNC interface to an external non-HEIDENHAIN device, use the following setting:
8 data bits, BCC any character, transmission stop with DC3, even character parity, character parity desired, 2 stop bits.

Input value: 1+0+8+0+32+64 = 105, so enter 105 for MP 5020.

Interface type

MP5030

Function	Selections	Value
• Interface type	Standard	0
	Interface for blockwise transfer	1

Parameters for 3D touch probes

Signal transmission type

MP6010

Function	Value
• Cable transmission	0
• Infrared transmission	1

Traversing behavior of touch probe

Parameter	Function	Value
MP6120	Probing feed rate in mm/min	80 to 30 000
MP6130	Maximum measuring range to first scanning point in mm	0 to 30 000
MP6140	Safety clearance over probing point during automatic probing, in mm	0 to 30 000
MP6150	Rapid traverse for probe cycle in mm/min	80 to 30 000

Parameters for TNC displays and the editor

Programming station

MP7210

Function	Value
• TNC with machine	0
• TNC as programming station with active PLC	1
• TNC as programming station with inactive PLC	2

Block number increment with ISO programming

MP7220

Function	Value
• Block number increment	0 to 255

Dialog language

MP7230

Function	Value
• National dialog language	0
• Dialog language English (standard)	1

Edit-protect OEM cycles

For protection against editing of programs whose program number is the same as an OEM cycle number.

MP7240

Function	Value
• Edit-protect OEM cycles	0
• No edit protection of OEM cycles	1

Defining a tool table (program 0)

Input: numerical value

Parameter	Function	Value
• **MP7260**	Total number of tools in the table	0 to 99
• **MP7261**	Number of tools with pocket numbers	0 to 99
• **MP 7264**	Number of reserved pockets next to special tools	0 to 3

Settings for MANUAL OPERATION mode

Entry values 0 to 3:
Sum of the individual values from the “value” column.

MP7270

Function	Selections	Value
• Display feed rate in manual mode	Display feed rate Do not display feed rate	+1 +0
• Spindle speed S and M functions still active after STOP	S and M still active S and M no longer active	+0 +2

Decimal character

MP7280

Function	Value
• Decimal point	1
• Decimal comma	0

Display steps for coordinate axes

MP7290

Function	Value
• Display step 0.001 mm	0
• Display step 0.005 mm	1

Q parameters and status display

MP7300

Function	Selections	Value
• Q parameters and status display	Do not erase Erase with M02, M30 and N9999	+0 +1
• Last programmed tool after power interruption	Do not activate Activate	+0 +4

Graphics display

Entry range: 0 to 3 (sum of the individual values)

MP7310

Function	Selections	Value
• View in 3 planes according to ISO 6433, Part 1	Projection method 1 Projection method 2	+0 +1
• Rotate coordinate system by 90° in the working plane	Rotate Do not rotate	+2 +0

Parameters for machining and program run

Effect of cycle G72 SCALING

MP7410

Function	Value
• SCALING effective in 3 axes	0
• SCALING effective in the working plane	1

MP7411 Tool compensation data in the TOUCH PROBE block

Function	Value
• Overwrite current tool data with the calibrated data of the touch probe	0
• Retain current tool data	1

Behavior of machining cycles

This general user parameter affects pocket milling.

Entry value: 0 to 15 (sum of the individual values in the "value" column)

MP7420

Function	Selections	Value
• Milling direction for a channel around the contour	Clockwise for pockets, counterclockwise for islands	+1
	Counterclockwise for pockets, clockwise for islands	+0
• Sequence of roughing out and channel milling	First mill contour channel, then rough out	+0
	First rough out, then mill contour channel	+2
• Merge contours	Merge compensated contours	+0
	Merge uncompensated contours	+4
• Milling in depth	At each pecking depth, mill channel and rough out before going to next depth	+8
	Mill contour channel to full pocket depth, then rough out to full pocket depth	+0

Overlapping with pocket milling

Overlap factor with pocket milling:
product of MP7430 and the tool radius

MP7430

Function	Value
• Overlap factor for pockets	0.1 to 1 414

Effect of M functions

The M functions M6 and M89 are influenced by MP 7440:

Entry range: 0 to 7
(Sum of the individual values in the "value" column)

MP7440

Function	Selections	Value
• Programmable stop with M06	Program stop with M06	+0
	No program stop with M06	+1
• Modal cycle call with M89	Modal cycle call with M89	+2
	M89 vacant M function	+0
• Axes are stopped when M function is carried out	Axis stop with M functions	+4
	No axis stop	+0

Safety limit for machining corners at constant path speed

Corners whose inside angle is less than the entered value are no longer machined at constant path speed with M90.

MP7460

Function	Value
• Maintain constant path speed at inside corners for angles of (degrees)	0 to 179.999

Coordinate display for rotary axis

MP7470

Function	Value
• Angle display up to ± 359.999°	0
• Angle display up to ± 30 000°	1

Parameters for override behavior and electronic handwheel

Override

Entry range: 0 to 7 (sum of the individual values in the “value” column)

MP7620

Function	Selections	Value
• Feed rate override when rapid traverse key pressed in program run mode	Override effective Override not effective	+1 +0
• Feed rate override when rapid traverse key and machine axis direction button pressed	Override effective Override not effective	+4 +0
• Increments for overrides	1% increments 0.01% increments	+0 +8

Setting the TNC for handwheel operation

Entry range: 0 to 5

MP7640

Function	Value
• No handwheel	0
• HR 330 with additional keys – the keys for traverse direction and rapid traverse are evaluated by the NC	1
• HR 130 without additional keys	2
• HR 330 with additional keys – the keys for traverse direction and rapid traverse are evaluated by the PLC	3
• HR 332 with 12 additional keys	4
• Multi-axis handwheel with additional keys	5

11.2 Miscellaneous Functions (M Functions)

Miscellaneous functions with predetermined effect

M	Function	Effective at	
		start of block	end of block
M00	Stop program run / Spindle stop / Coolant off		•
M02	Stop program run / Spindle stop / Coolant off. Clear the status display (depending on machine parameter) / Return to block 1		•
M03	Spindle on clockwise	•	
M04	Spindle on counterclockwise	•	
M05	Spindle stop		•
M06	Tool change / Stop program run (depending on machine parameter) / Spindle stop		•
M08	Coolant on	•	
M09	Coolant off		•
M13	Spindle on clockwise / Coolant on	•	
M14	Spindle on counterclockwise / Coolant on	•	
M30	Same function as M02		•
M89	Vacant miscellaneous function	•	
	or		
	Cycle call, modally effective (depending on machine parameter)		•
M90	Smoothing corners	•	
M91	**Within the positioning block:** Coordinates are referenced to the machine datum	•	
M92	**Within the positioning block:** Coordinates are referenced to a position defined by the machine tool builder (such as a tool change position)	•	
M93	**Within the positioning block:** Coordinates are referenced to the current tool position. Effective in blocks with R0, R+, R–	•	
M94	Reduce display of rotary axis to a value under 360°	•	
M95	Reserved		•
M96	Reserved		•
M97	Machine small contour steps		•
M98	Completely machine open contours		•
M99	Blockwise cycle call		•

Vacant miscellaneous functions

Vacant M functions are defined by the machine tool builder. They are described in the operating manual of your machine tool.

Effect of vacant miscellaneous functions

M	Function	Effective at	
		start of block	end of block
M01			•
M07		•	
M10			•
M11		•	
M12			•
M15		•	
M16		•	
M17		•	
M18		•	
M19			•
M20		•	
M21		•	
M22		•	
M23		•	
M24		•	
M25		•	
M26		•	
M27		•	
M28		•	
M29		•	
M31		•	
M32			•
M33			•
M34			•
M35			•
M36		•	
M37		•	
M38		•	
M39		•	
M40		•	
M41		•	
M42		•	
M43		•	
M44		•	
M45		•	
M46		•	
M47		•	
M48		•	
M49		•	

M	Function	Effective at	
		start of block	end of block
M50		•	
M51		•	
M52			•
M53			•
M54			•
M55		•	
M56		•	
M57		•	
M58		•	
M59		•	
M60			•
M61		•	
M62		•	
M63			•
M64			•
M65			•
M66			•
M67			•
M68			•
M69			•
M70			•
M71		•	
M72		•	
M73		•	
M74		•	
M75		•	
M76		•	
M77		•	
M78		•	
M79		•	
M80		•	
M81		•	
M82		•	
M83		•	
M84		•	
M85		•	
M86		•	
M87		•	
M88		•	

11.3 Preassigned Q Parameters

The Q parameters Q100 to Q113 are assigned values by the TNC. Such values include:

- Values from the PLC
- Tool and spindle data
- Data on operating status, etc.

Values from the PLC: Q100 to Q107

The TNC uses the parameters Q100 to Q107 to transfer values from the PLC to an NC program.

Tool radius: Q108

The radius of the current tool is assigned to Q108.

Tool axis: Q109

The value of parameter Q109 depends on the current tool axis.

Tool axis	Parameter value
No tool axis defined	Q109 = –1
Z axis	Q109 = 2
Y axis	Q109 = 1
X axis	Q109 = 0

Spindle status: Q110

The value of Q110 depends on the M function last programmed for the spindle.

M function	Parameter value
No spindle status defined	Q110 = –1
M03: Spindle on clockwise	Q110 = 0
M04: Spindle on counterclockwise	Q110 = 1
M05 after M03	Q110 = 2
M05 after M04	Q110 = 3

Coolant on/off: Q111

M function	Parameter value
M08: Coolant on	Q111 = 1
M09: Coolant off	Q111 = 0

Overlap factor: Q112

The overlap factor for pocket milling (MP 7430) is assigned to Q112.

Unit of measurement: Q113

The value of parameter Q113 specifies whether the highest-level NC program (for nesting with %) is programmed in millimeters or inches. After NC start, Q113 is set as follows:

Unit of measurement (main program)	Parameter value
Millimeters	Q113 = 0
Inches	Q113 = 1

Current tool length: Q114

The current value of the tool length is assigned to Q114.

Coordinates from probing during program run

Parameters Q115 to Q118 are assigned the coordinates of the spindle position upon probing during a programmed measurement with the 3D touch probe. The length and radius of the stylus are not compensated for these coordinates.

Coordinate axis	Parameter
X axis	Q115
Y axis	Q116
Z axis	Q117
IV axis	Q118

Current tool radius compensation

The current tool radius compensation is assigned to parameter Q123 as follows:

Current tool compensation	Parameter value
R0	Q123 = 0
RL	Q123 = 1
RR	Q123 = 2
R+	Q123 = 3
R–	Q123 = 4

11.4 Diagrams for Machining

Spindle speed S

The spindle speed S can be calculated from the tool radius R and the cutting speed v as follows:

$$S = \frac{V}{2 \cdot R \cdot \pi}$$

Units:

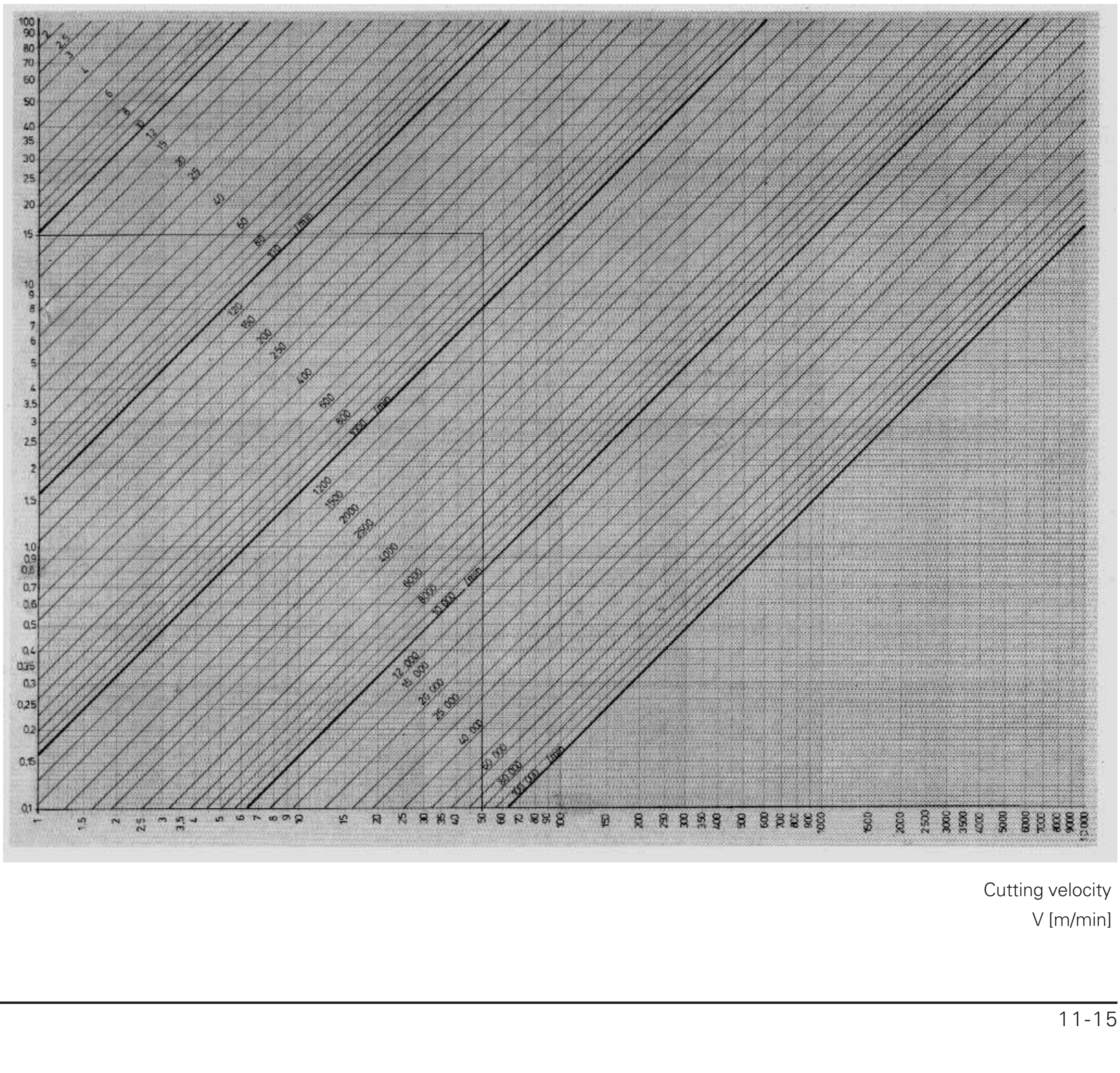
S in rpm
V in m/min
R in mm

You can read the spindle speed directly from the diagram.

Example:

Tool radius	R = 15 mm
Cutting speed	V = 50 m/min
Spindle speed	S ≈ 500 rpm (calculated S = 497 rpm)

Tool radius
R [mm]



Cutting velocity
V [m/min]

Feed rate F

The feed rate F of the tool is calculated from the number of tool teeth n, the permissible depth of cut per tooth d, and the spindle speed S:

F = n · d · S

Units:

F	in	mm/min
d	in	mm
S	in	rpm

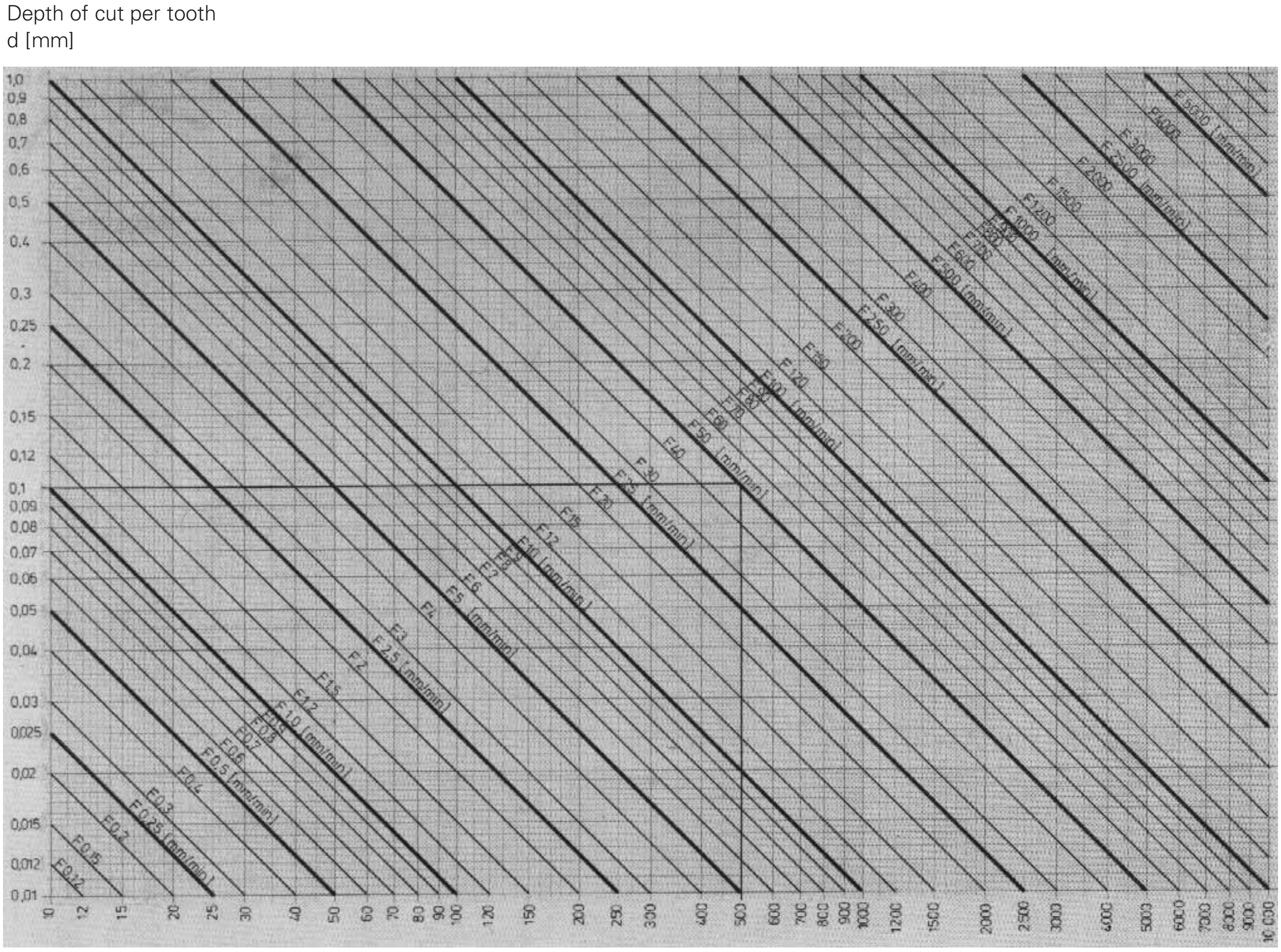
The feed rate read from the diagram must be multiplied by the number of tool teeth.

Example:

Depth of cut per tooth	d	=	0.1 mm
Spindle speed	S	=	500 rpm
Feed rate from diagram	F	=	50 mm/min
Number of tool teeth	n	=	6
Feed rate to enter	F	=	300 mm/min

The diagram provides approximate values and assumes the following:

- Downfeed in the tool axis = 0.5 . R and the tool is cutting through solid metal, or
- Lateral metal-to-air ratio = 0.25 . R and downfeed in the tool axis = R



Feed rate F for tapping

The feed rate for tapping F is calculated from the thread pitch p and the spindle speed S:

$F = p \cdot S$

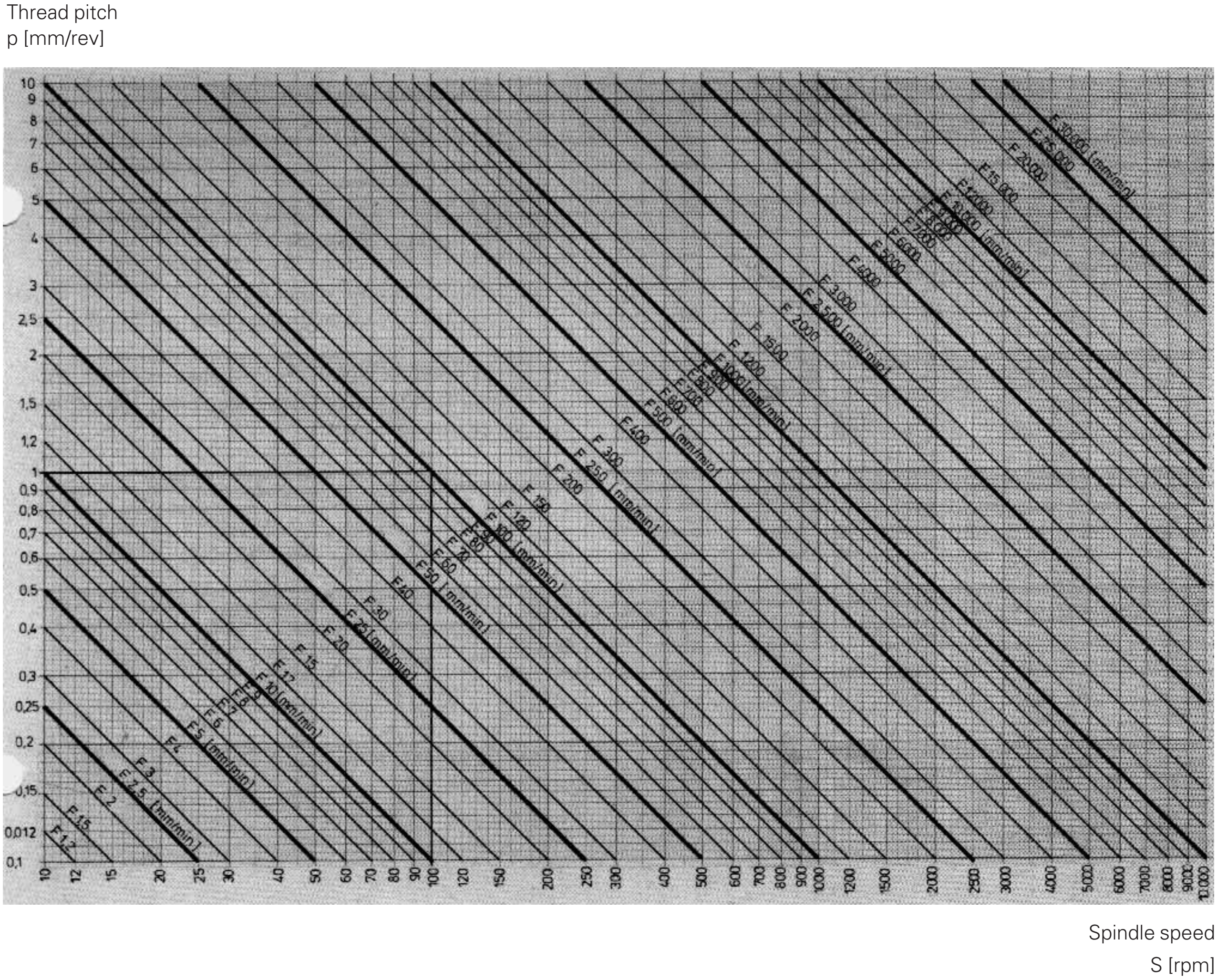
Units:

F	in	mm/min
p	in	mm/1
S	in	rpm

The feed rate for tapping can be read directly from the diagram below.

Example:

Thread pitch	p	=	1 mm/rev
Spindle speed	S	=	100 rpm
Feed rate for tapping	F	=	100 mm/min



11.5 Features, Specifications and Accessories

TNC 360

Description

Contouring control for up to 4 axes, with oriented spindle stop.

Components

Logic unit, keyboard, monochrome flat luminescent screen or CRT.

Data interface

RS-232-C / V.24

Simultaneous axis control for contour elements

• Straight lines up to	3 axes
• Circles in	2 axes
• Helices	3 axes

Background programming

For editing one part program while the TNC is running another.

Test run

Internally and with test run graphics.

Program types

- HEIDENHAIN plain language format and ISO programs
- Tool table (program 0)

Program memory

- Battery-buffered for up to 32 programs
- Capacity: approximately 4000 program blocks

Tool definitions

- Up to 254 tools in one program or up to 99 tools in the tool table (program 0).

Programmable Functions

Contour elements

Straight line, chamfer, circular arc, circle center, circle radius, tangentially connecting arc, corner rounding.

Program jumps

Subprogram, program section repeat, main program as subprogram.

Fixed cycles

Pecking, tapping (also with synchronized spindle), rectangular and circular pocket milling, slot milling, milling pockets and islands from a list of subcontour elements.

Coordinate transformations

Datum shift, mirroring, rotation, scaling factor.

3D Touch Probe System

Probing functions for measuring and datum setting, digitizing 3D surfaces (optional, only available with HEIDENHAIN plain language programming).

Mathematical functions

Basic operations +, –, · and %, trigonometric functions
sin, cos, tan and arctan.
Square roots ($\sqrt{a}$) and root sum of squares ($\sqrt{a^2 + b^2}$).
Logical comparisons greater than, smaller than, equal to, not equal to.

TNC Specifications

Block execution time	1500 blocks/min (40 ms per block)
Control loop cycle time	6 ms
Data transfer rate	Max. 38400 baud
Ambient temperature	0°C to 45°C (operation) –30°C to 70°C (storage)
Traverse	Max. ± 30 m (1181 inches)
Traversing speed	Max. 30 m/min (1181 ipm)
Spindle speed	Max. 99 999 rpm
Input resolution	As fine as 1 µm (0.0001 in.) or 0.001°

Accessories

FE 401 Floppy Disk Unit

Description	Portable bench-top unit
Applications	All TNC contouring controls, TNC 131, TNC 135
Data interfaces	Two RS-232-C interface ports
Data transfer rate	• TNC : 2400 to 38400 baud • PRT : 110 to 9600 baud
Diskette drives	Two drives, one for copying, capacity 795 kilobytes (approx. 25 000 blocks), up to 256 files
Diskette type	3.5", DS DD, 135 TPI

Triggering 3D Touch Probes

Description	Touch probe system with ruby tip and stylus with rated break point, standard shank for spindle insertion
Models	TS 120: Cable transmission, integrated interface TS 511: Infrared transmission, separate transmitting and receiving units
Spindle insertion	TS 120: manual TS 511: automatic
Probing reproducibility	Better than 1 µm (0.000 04 in.)
Probing speed	Max. 3 m/min (118 ipm)

Electronic Handwheels

HR 130	• Integrable unit
HR 330	• Portable version with cable transmission, equipped with axis address keys, rapid traverse key, safety switch, emergency stop button.

11.6 TNC Error Messages

The TNC automatically generates error messages when it detects such things as

- Incorrect data input
- Logical errors in the program
- Contour elements that are impossible to machine
- Incorrect use of the touch probe system

An error message containing a program block number was caused by an error in that block or in the preceding block. To clear an error message, first correct the problem and then press the CE key.

Some of the more frequent error messages are explained in the following list.

TNC error messages during programming

BLOCK NUMBER ALLOCATED

Assign a new block number with N that has not been used yet in the program.

ENTRY VALUE INCORRECT

- Enter a correct LABEL number.
- Press the correct key.

EXT. IN-/OUTPUT NOT READY

The external device is not correctly connected.

FURTHER PROGRAM ENTRY IMPOSSIBLE

Erase some old files to make room for new ones.

JUMP TO LABEL 0 NOT PERMITTED

Do not program L 0,0.

LABEL NUMBER ALLOCATED

Label numbers can only be assigned once.

TNC error messages during test run and program run

ANGLE REFERENCE MISSING

- Define the arc and its end points unambiguously.
- If you enter polar coordinates, define the polar coordinate angle correctly.

ARITHMETICAL ERROR

You have attempted to calculate with illegal values.

- Define values within the range limits.
- Choose probe positions for the 3D touch probe that are farther separated.
- Calculations must be mathematically possible.

AXIS DOUBLE PROGRAMMED

Each axis can only have one value for position coordinates.

BLK FORM DEFINITION INCORRECT

- Program the MIN and MAX points according to the instructions.
- Choose a ratio of sides less than 84:1.
- When programming with %, copy G30/G31 into the main program.

CHAMFER NOT PERMITTED

- A chamfer block must be inserted between two straight line blocks with the same radius compensation.
- Do not change the program during program run.
- Do not edit the program while it is transferred or executed.

CIRCLE END POS. INCORRECT

- Enter complete information for tangential arcs.
- Enter end points that lie on the circular path.

CYCL INCOMPLETE

- Define the cycle with all data in the proper sequence.
- Do not call coordinate transformation cycles.
- Define a cycle before calling it.
- Enter a pecking depth other than 0.

EXCESSIVE SUBPROGRAMMING

- Conclude subprograms with G98 L0.
- Program subprogram calls without repetition (L ..,0).
- Program a call for program section repeats to include the repetitions (L ..,5).
- Subprograms cannot call themselves.
- Subprograms cannot be nested in more than 8 levels.
- Main programs cannot be nested as subprograms in more than 4 levels.

FEED RATE IS MISSING

- Enter the feed rate for the positioning block.
- Enter FMAX in each block.

GROSS POSITIONING ERROR

The TNC monitors positions and movements. If the actual position deviates to greatly from the nominal position, this blinking error message is displayed. Press the END key for a few seconds to correct this error (warm start).

KEY NON-FUNCTIONAL

This message always appears when you press a key that is not needed for the current dialog.

LABEL NUMBER NOT ALLOCATED

You can only call labels numbers that have been assigned.

PATH OFFSET WRONGLY ENDED

Do not cancel tool radius compensation in a block with a circular path.

PATH OFFSET WRONGLY STARTED

- Use the same radius compensation before and after a G24 and G25 block.
- Do not begin tool radius compensation in a block with a circular path.

PGM SECTION CANNOT BE SHOWN

- Enter a smaller tool radius.
- Movements in a rotary axis cannot be graphically simulated.
- Enter a tool axis for simulation that is the same as the axis in block G30.

PLANE WRONGLY DEFINED

- Do not change the tool axis while a basic rotation is active.
- Define the main axes for circular arcs correctly.
- Define both main axes for I, J, K.

PROBE SYSTEM NOT READY

- Orient transmitting/receiving window of TS 511 to face receiving unit.
- Check whether the touch probe is ready for operation.

PROGRAM-START UNDEFINED

- Program the first traverse block with G00 G90 G40 (tool must be called previously).
- Do not resume an interrupted program at a block with a tangential arc or pole transfer.

RADIUS COMPENSATION UNDEFINED

Enter radius compensation in the first subprogram to cycle G37:
CONTOUR GEOM.

ROUNDING OFF NOT DEFINED

Enter tangentially connecting arcs and rounding arcs correctly.

ROUNDING RADIUS TOO LARGE

Rounding arcs must fit between contour elements.

SELECTED BLOCK NOT ADDRESSED

Before a test run or program run you must go to the beginning of the program by entering GOTO 0.

STYLUS ALREADY IN CONTACT

Before probing, pre-position the stylus so that it is not touching the workpiece surface.

TOOL RADIUS TOO LARGE

Enter a tool radius that

- lies within the given limits, and
- permits the contour elements to be calculated and machined.

TOUCH POINT INACCESSIBLE

Pre-position the 3D touch probe to a point nearer the surface.

WRONG AXIS PROGRAMMED

- Do not attempt to program locked axes.
- Program a rectangular pocket or slot in the working plane.
- Do not mirror rotary axes.
- Chamfer length must be positive.

WRONG RPM

Program a spindle speed within the permissible range.

WRONG SIGN PROGRAMMED

Enter the correct sign for the cycle parameter.

11.7 Address letters (ISO programming)

G Functions

Group	G	Function	Effective blockwise	Refer to page
Positioning functions	00	Linear interpolation, Cartesian coordinates, at rapid traverse		5-10
	01	Linear interpolation, Cartesian coordinates		5-10
	02	Circular interpolation, Cartesian coordinates, clockwise		5-18
	03	Circular interpolation, Cartesian coordinates, counterclockwise		5-18
	05	Circular interpolation, Cartesian coordinates, no direction of rotation defined		5-18
	06	Circular interpolation, Cartesian coordinates, tangential connection		5-24
	07	Single axis positioning block	*	5-41
	10	Linear interpolation, polar coordinates, at rapid traverse		5-28
	11	Linear interpolation, polar coordinates		5-28
	12	Circular interpolation, polar coordinates, clockwise		5-30
	13	Circular interpolation, polar coordinates, counterclockwise		5-30
	15	Circular interpolation, polar coordinates, no direction of rotation defined		5-30
	16	Circular interpolation, polar coordinates, tangential connection		5-32
Cycles	04	Dwell time	*	8-38
	28	Mirror image		8-33
	36	Oriented spindle stop		8-39
	37	Definition of the pocket contour		8-26
	39	Cycle for program call, cycle call with G79	*	8-3
	54	Datum shift in a part program		8-30
	56	Pilot drilling contour pockets (combined with G37)		8-25
	57	Roughing out contour pockets (combined with G37)		8-17
	58	Contour milling, clockwise (combined with G37)		8-26
	59	Contour milling, counterclockwise (combined with G37)		8-26
	72	Scaling factor		8-36
	73	Rotation of the coordinate system		8-35
	74	Slot milling		8-9
	75	Rectangular pocket milling, clockwise		8-11
	76	Rectangular pocket milling, counterclockwise		8-11
	77	Circular pocket milling, clockwise		8-13
	78	Circular pocket milling, counterclockwise		8-13
	83	Pecking		8-4
	84	Tapping with a floating tap holder		8-6
	85	Rigid tapping		8-8
	79	Cycle call	*	8-3
Selecting the working plane	17	Select plane XY, tool axis Z		5-16
	18	Select plane ZX, tool axis Y		5-16
	19	Select plane YZ, tool axis X		5-16
	20	Tool axis IV		5-16
Chamfers	24	Chamfer with chamfer length R	*	5-13
Corner rounding	25	Corner rounding with radius R	*	5-26
Approaching and departing a contour	26	Smooth approach of a contour with radius R	*	5-6
	27	Smooth departure from a contour with radius R	*	5-6
	29	Define the last programmed position as a pole		5-28
Definition of the workpiece blank	30	Define the blank form for graphic simulation, MIN point		4-14
	31	Define the blank form for graphic simulation, MAX point		4-14
	38	Stop program run	*	3-4
Traverse with/without radius compensation	40	No tool compensation (R0)		4-12
	41	Tool radius compensation, tool traverse to the left of the contour (RL)		4-12
	42	Tool radius compensation, tool traverse to the right of the contour (RR)		4-12
	43	Lengthening single-axis movements (R+)		4-12
	44	Shortening single-axis movements (R–)		4-12
	50	Edit protection at the beginning of a program		1-20
	51	Next tool number (with central tool memory)	*	4-9
	55	Touch probe function	*	7-17
Unit of measurement	70	Unit of measurement: Inches (at beginning of program)		4-14
	71	Unit of measurement: Millimeters (at beginning of program)		4-14
Definition of positions	90	Absolute workpiece positions		1-11
	91	Incremental workpiece positions		1-11
	98	Assigning a label number	*	6-2
	99	Tool definition	*	4-7

Other address letters

Address letter	Function
%	Begin program or call program with G39
A **B** **C**	Rotate around the X axis Rotate around the Y axis Rotate around the Z axis
D	Parameter definition (program parameter Q)
F **F** **F**	Feed rate Dwell time with G04 Scaling factor with G72
G	Traversing conditions
H **H**	Polar angle in chain dimensions/absolute dimensions Rotation angle with G73
I **J** **K**	X coordinate of circle center/pole Y coordinate of circle center/pole Z coordinate of circle center/pole
L **L** **L**	Assign a label number with G98 Jump to a label number Tool length with G99
M	Help functions
N	Block number
P **P**	Cycle parameters in fixed cycles Parameters in parameter definitions
Q	Program parameter/Cycle parameter Q
R **R** **R** **R** **R**	Polar radius Circle radius with G02/G03/G05 Rounding radius with G25/G26/G27 Chamfer side length with G24 Tool radius with G99
S **S**	Spindle speed Oriented spindle stop with G36
T **T**	Tool definition with G99 Tool call
U **V** **W**	Linear movement parallel to the X axis Linear movement parallel to the Y axis Linear movement parallel to the Z axis
X **Y** **Z**	X axis Y axis Z axis
*****	End of block

Parameter definitions

D	Function	Refer to page
00	Assign	7-3
01 02 03 04	Addition Subtraction Multiplication Division	7-5 7-5 7-5 7-5
05	Square root	7-5
06 07	Sine Cosine	7-7 7-7
08	Root sum of squares ($c = \sqrt{a^2 + b^2}$)	7-7
09 10 11 12	If equal, jump If not equal, jump If greater than, jump If less than, jump	7-8 7-8 7-8 7-8
13	Angle (calculated from c · sin ð and c · cos ð)	7-7
14	Error number	7-11
15	Print	7-11
19	Assign values for the PLC	7-11

Miscellaneous Functions (M Functions)
Miscellaneous functions with predetermined effect

M	Function	Effective at start of block	 end of block
M00	Stop program run / Spindle stop / Coolant off		•
M02	Stop program run / Spindle stop / Coolant off. Clear the status display (depending on machine parameter) / Return to block 1		•
M03	Spindle on clockwise	•	
M04	Spindle on counterclockwise	•	
M05	Spindle stop		•
M06	Tool change / Stop program run (depending on machine parameter) / Spindle stop		•
M08	Coolant on	•	
M09	Coolant off		•
M13	Spindle on clockwise / Coolant on	•	
M14	Spindle on counterclockwise / Coolant on	•	
M30	Same function as M02		•
M89	Vacant miscellaneous function	•	
	or Cycle call, modally effective (depending on machine parameter)		•
M90	Smoothing corners	•	
M91	**Within the positioning block:** Coordinates are referenced to the machine datum	•	
M92	**Within the positioning block:** Coordinates are referenced to a position defined by the machine tool builder (such as a tool change position)	•	
M93	**Within the positioning block:** Coordinates are referenced to the current tool position. Effective in blocks with R0, R+, R–	•	
M94	Limit display of rotary axis to value under 360°	•	
M95	Reserved		•
M96	Reserved		•
M97	Machine small contour steps		•
M98	Completely machine open contours		•
M99	Blockwise cycle call		•